MONARCH-E

PCI Frame Synchronizer
Telemetry Simulator
with Reed-Solomon
Encoder/Decoder

Functional Description

Version: 1.0

Date Prepared: May 16, 2002

Export Controls: Avtec's products and technology are controlled for export purposes by the U.S. Government pursuant to the Arms Export Control Act and the International Traffic in Arms Regulations or the Export Administration Act and the Export Administration Regulations. It is the company's responsibility to comply with the applicable export laws and regulations before engaging in export activities, including engaging in exports, reexports, or the disclosure of technical data in the U.S. to foreign persons, a deemed export.

TM-02-019-00

AVTEC SYSTEMS

MONARCH-E
PCI Frame Synchronizer Telemetry Simulator with Reed-Solomon Encoder/Decoder

Functional Description

Support

Technical Support

This manual provides engineers with integration and programming information necessary to develop a system using the MONARCH-E. The MONARCH-E is a PCI-based serial I/O card that supports both CCSDS and TDM data formats. MONARCH-E Technical Support is available from AVTEC Systems, Inc.

Avtec Systems, Inc.
Attn: MONARCH-E Technical Support
10530 Rosehaven St., Suite 300
Fairfax, VA 22030-2840
(703) 273-2211
support@avtec.com

Customer Service

Direct non-technical questions and business related matters to Customer Support.

REVISION HISTORY

REV.	DESCRIPTION	DATE	APPD.
-00	Baseline release. Added Xilinx configuration, PLX configuration, Appended Status, DMA transfer sections.	5-16-02	5-21-02

TABLE OF CONTENTS

FIGURES

TABLES

Chapter 1
Introduction

Section 1.1
OVERVIEW

The MONARCH-E Frame Synchronizer/PCM Simulator by Avtec Systems, Inc., is a PCI-based serial I/O card that supports both CCSDS (Conventional and Advanced Orbiting Systems) and time division multiplexed (TDM) data formats. The MONARCH-E also provides a Reed-Solomon Encoder/Decoder to provide complete block error detection and correction for each of the CCSDS-recommended grades of service for Advanced Orbiting Systems.

The MONARCH-E input channel accepts serial data and clock at up to 25 Mbps from a bit synchronizer with Viterbi decoder and outputs frame data to the PCI bus. The input channel performs frame synchronization, derandomization, CRC error detection, de-interleaving, Reed-Solomon error detection and correction, time-tagging, and quality annotation. The input channel transfers annotated frame data to host computer memory for further processing. The MONARCH-E output channel reads frame data from host computer memory via the PCI bus and outputs serial data and clock at up to 25 Mbps. The output channel performs Reed-Solomon encoding, interleaving, CRC encoding, pseudo-randomization, convolutional encoding, and error insertion.

Section 1.2
SYSTEM ARCHITECTURE

The block diagram describing the MONARCH-E System Architecture is shown in **Figure 1-1**. The MONARCH-E contains six functional components:

- Serial Input Logic
- Reed-Solomon Error Correction chip
- PCI Interface
- Reed-Solomon Encoder
- Serial Output Logic
- Frequency Synthesizer

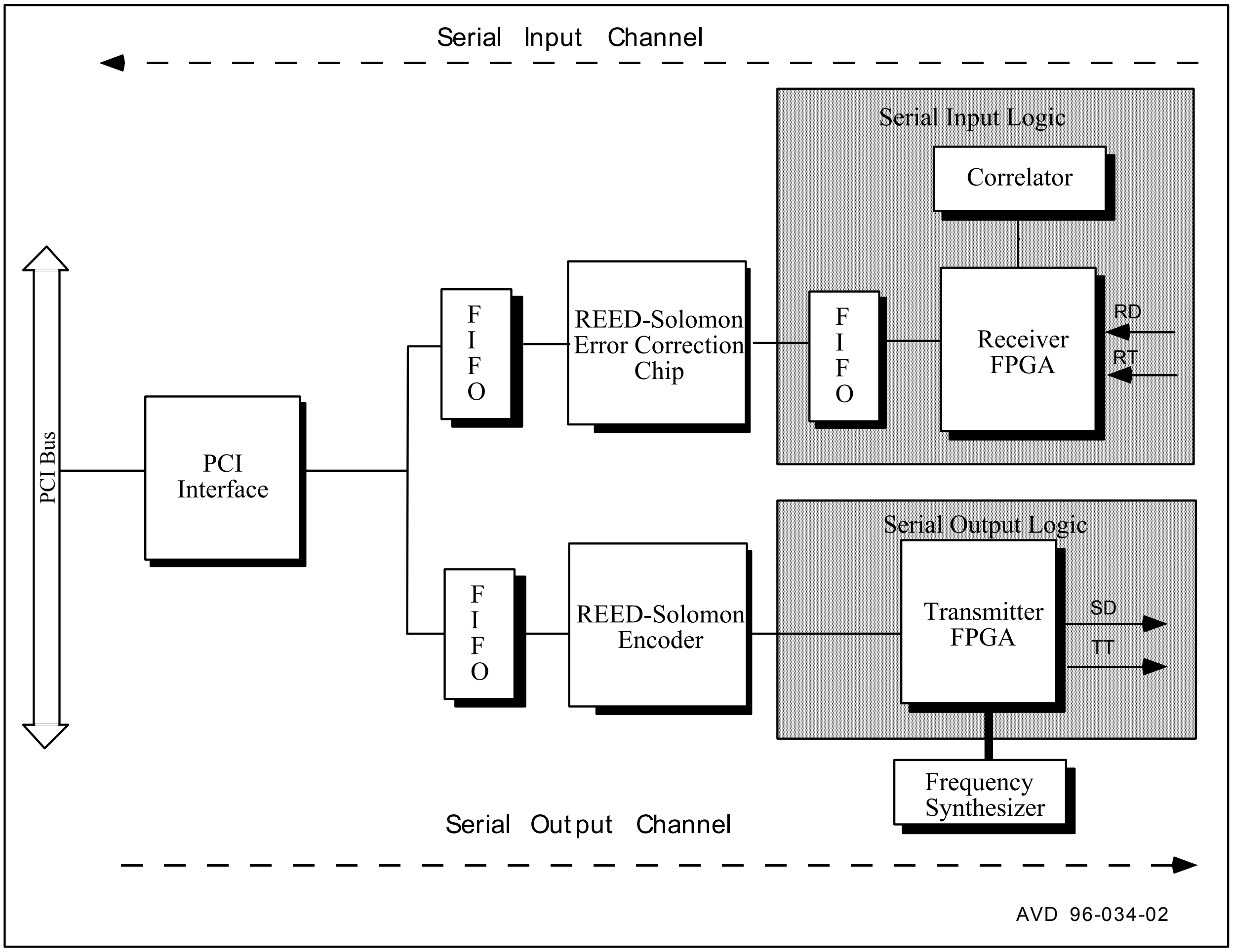


Figure 1-1: MONARCH-E System Block Diagram

SERIAL INPUT LOGIC

The serial input logic processes both CCSDS and TDM data formats. For CCSDS telemetry, the serial input logic synchronizes to channel access data units and formats them for input to the Reed-Solomon Decoder. For TDM telemetry, the serial input logic bypasses the Reed-Solomon Decoder and outputs frame data directly to the PCI interface. Programmable frame synchronizer parameters include: sync pattern and mask, error threshold, check frames, flywheel frames, frame length, word size, and bit slip window. The serial input logic automatically performs polarity correction based on the detected sync pattern polarity. The serial input logic also performs derandomization and CRC-16 VCDU error detection, and can also append data quality information and a 32-bit time tag to frame of data.

REED-SOLOMON ERROR CORRECTION (RSEC)

The RSEC chip is an LSI Logic ASIC that was designed by engineers at NASA's Goddard Space Flight Center. The RSEC chip accepts parallel data from the Serial Input Logic. It performs RS(255,223) VCDU error correction with interleaving depth from 1 to 8, RS(10,6) VCDU header error correction, and real-time quality generation and annotation. Shortened code blocks are also supported. The RSEC chip outputs corrected VCDUs with quality annotation to the PCI Interface.

PCI Interface

The PCI interface is based on the PLX PCI9080 chip. It provides full PCI compatibility, including support for burst mode operation with any PCI target. The PCI interface transfers frame data to/from the host processor's memory at up to 132 MB/sec.

Reed Solomon Encoder

The Reed-Solomon Encoder is implemented in a Field Programmable Gate Array (FPGA). The encoder accepts parallel data from the PCI interface and performs RS(255,223) VCDU encoding with an interleave depth from 1 to 8. Shortened code blocks are also supported. The Reed-Solomon Encoder outputs interleaved code words to the Serial Output Logic.

Serial Output Logic

The serial output logic performs parallel-to-serial conversion, CRC encoding, pseudo-randomization, convolution encoding, and bit error burst insertion at up to 25 Mbps

Frequency Synthesizer

The MONARCH-E contains two transmit clock frequency synthesizers, a Direct Digital Synthesizer (DDS) and a Phased Lock Loop (PLL)-based programmable clock chip. The DDS provides very high resolution frequency output from 10 Hz to 10 MHz. The programmable clock provides a lower resolution frequency output from 320 kHz to 100 MHz.

Section 1.3
REFERENCES

The following publications were used in the preparation of the MONARCH-E Functional Description:

1) CCSDS 101.0-B-3 Consultative Committee for Space Data Systems, Recommendation for Space Data System Standards Telemetry Channel Coding, CCSDS 101.0-B-3, May 1992.

2) 521-SPEC-002 MSB ASIC Components Document, Volume 1, Section 10, Reed-Solomon Error Correction Chip, GSFC Microelectronics Systems Branch

3) PCI9080 Data Sheet, (PLX Technologies, www.plxtech.com), July 1997.

4) ICD2051 Data Sheet: PLL Clock Generator (Cypress, www.cypress.com), April 1995.

5) HSP45102 Data Sheet: DDS Clock Generator (Intersil, www.intersil.com), January 1994.

6) PCI Local Bus Specification Revision 2.0 (PCI SIG, www.pcisig.com), PCI Special Interest Group, April 1993.

Chapter 2
Specifications

Section 2.1
RECEIVER

FRAME SYNCHRONIZER

- Programmable frame sync pattern up to 64 bits
- Programmable frame sync mask up to 64 bits
- Programmable error threshold for acceptable sync patterns up to 15 bit errors
- Adaptive sync strategy with 0 to 7 check frames and 0 to 7 flywheel frames
- Programmable bit slip window from 0 to ± 3 bits
- Auto-polarity detection and correction
- Frame length up to 64K words/frame
- Programmable word size from 4 to 16 bits

DE-RANDOMIZER

- Exclusive OR received frame data following sync pattern with pseudo-random pattern given by $h(x) = x^8 + x^7 + x^5 + x^3 + 1$
- Shift register is initialized to all "ones" at the start of each frame
- Programmable start position

CRC FRAME ERROR DETECTION

- Compute frame error control field from received data using the polynomial $g(x) = x^{16} + x^{12} + x^5 + 1$
- Feed-back shift register is initialized to all "ones" at the start of each frame
- Programmable start and end of included data

REED-SOLOMON ERROR CORRECTION

- CCSDS Reed-Solomon (255,223) error correction
- Support for shortened code blocks using "virtual" fill
- Interleave depth from 1 to 8
- CCSDS Reed-Solomon (10,6) header error correction
- Real-time quality generation and annotation for each VCDU

Section 2.2
TRANSMITTER

SERIAL OUTPUT LOGIC

- Programmable clock and data polarity
- Frame length up to 64K words/frame
- Programmable word size from 4 to 16 bits
- Programmable output frequency from 100 Hz to 25 MHz
- Programmable bit error burst insertion

CRC FRAME ERROR CONTROL

- Compute frame error control field using the polynomial $g(x) = x^{16} + x^{12} + x^5 + 1$
- Feed-back shift register is initialized to all "ones" at the start of each frame
- Overlay computed CRC remainder into output frame
- Programmable start and end of included data

REED-SOLOMON ENCODER

- CCSDS Reed-Solomon (255,223) encoding
- Support for shortened code blocks using "virtual" fill
- Interleave depth from 1 to 8

PSEUDO-RANDOMIZER

- Exclusive OR frame data following sync pattern with pseudo-random pattern given by $h(x) = x^8 + x^7 + x^5 + x^3 + 1$
- Shift register is initialized to all "ones" at the start of each frame
- Programmable start position

CONVOLUTIONAL ENCODER

- Rate 1/2, constraint-length 7, convolutional code
- Connection vectors G1 = 1111001 and G2 = 1011011
- Programmable parity order and parity inversion

FREQUENCY SYNTHESIZERS

- Direct digital synthesizer with output frequency from 10 Hz to 10 MHz
- PLL based programmable clock generator with output frequency from 320 kHz to 100 MHz

Section 2.3
SERIAL INPUT/OUTPUT OPTIONS

- Data rates to 25 Mbps
- TLL (50 Ohm termination) or RS-422 (124 Ohm termination) clock and data signals
- Female DB-37 connector for clock and data signals
- Selectable internal or external transmit clock
- Supports NRZ-L, S, M and Biphase-L, S, M data codes (Biphase is only for transmitter)
- Programmable clock and data polarity
- Breakout cable with BNC connectors for TTL signals
- Breakout cable with TRIAX connectors for RS-422 signals

Section 2.4
PCI INTERFACE

- Based on PLX PCI9080
- 132 MB/s peak bandwidth with 33 MHz PCI bus
- Hidden central arbitration
- Full multi-master capability
- Address and data parity
- Support for auto-configuration

Section 2.5
PHYSICAL SPECIFICATIONS

The MONARCH-E physical specifications are listed in the table below.

Table 2-1: MONARCH-E Physical Specifications

Operating Temperature with forced air cooling	0^{o} to 50^{o} C
Storage Temperature	-40^{o} to 85^{o} C
Relative Humidity (non-condensing)	5% to 95%
Board Dimensions	4.2” X 7”
Power Dissipation (Max)	7.5 watts
Mean Time Between Failures (MTBF)	172,208 hours (19 years)

Chapter 3
Theory of Operation

The MONARCH-E's six functional components as listed in **System Architecture** (Section 1.2) are described in detail in the following paragraphs.

Section 3.1
SERIAL INPUT LOGIC

The MONARCH-E input channel accepts serial data and clock from a bit synchronizer with Viterbi decoder and outputs frame data to the PCI bus. The input channel performs frame synchronization, derandomization, CRC error detection, de-interleaving, Reed-Solomon error detection and correction, time-tagging, and quality annotation. The input channel transfers annotated frame data to PC memory for further processing.

The MONARCH-E supports both TDM and CCSDS telemetry formats. For CCSDS telemetry, the serial input logic synchronizes to channel access data units and formats them for input to the Reed-Solomon Decoder and the Reed-Solomon Decoder transfers corrected frames to the PCI interface. For TDM telemetry, the serial input logic bypasses the Reed-Solomon Decoder and outputs frame data directly to the PCI interface.

The serial input logic performs frame synchronization using an adaptive strategy. Programmable frame synchronizer parameters include: sync pattern and mask, error threshold, check frames, flywheel frames, frame length, word size, and bit slip window. The serial input logic automatically performs polarity correction based on the detected sync pattern polarity. The serial input logic can also append data quality information and a 32-bit time tag to frame data.

A block diagram of the MONARCH-E serial input logic module is shown in **Figure 3-1**. The main blocks are the code converter, the pseudo-derandomizer, the CRC decoder, the bit counter and word counter, the input correlator, the frame synchronization logic, the serial-to-parallel shift register, FIFO buffers, the timer counter, and the interrupt control. All of the serial input logic except the FIFO buffers and the digital correlator are implemented in a single field programmable gate array (FPGA).

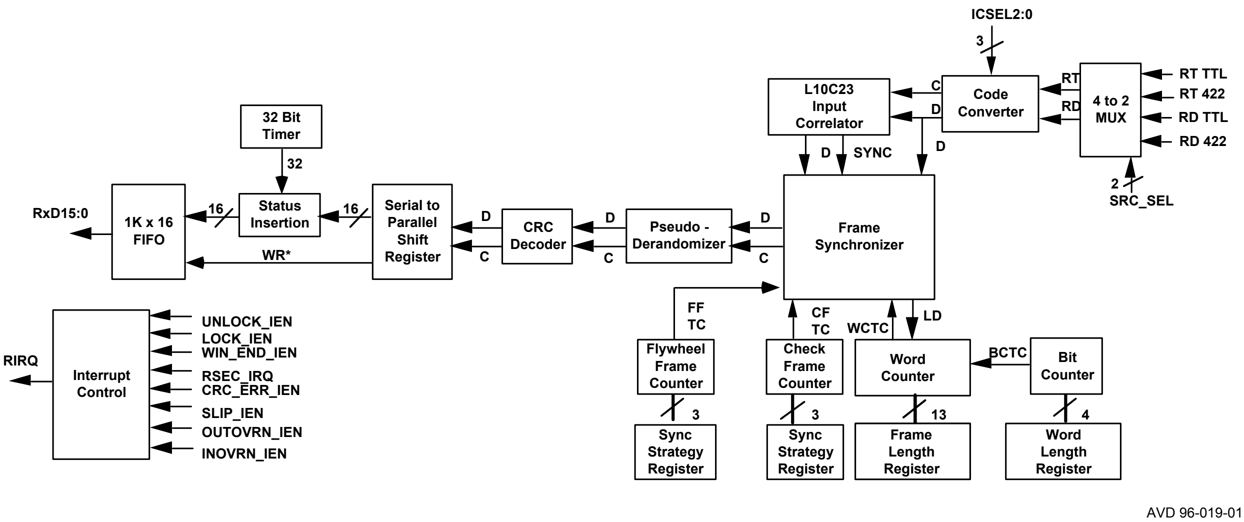


Figure 3-1: MONARCH-E Serial Input Logic

CODE CONVERTER

The code converter accepts NRZ-L, S, or M, or Biphase-L, S or M data from the onboard line receivers and outputs NRZ-L data. The conversion mode is software selectable.

PSEUDO-RANDOMIZER

The derandomizer receives serial data from the code converter. The derandomizer removes the randomized pattern to restore the original data in the transfer frame. The pseudo-random sequence is generated using the polynomial $h(x) = x^8 + x^7 + x^5 + x^3 + 1$. The sequence generator is re-initialized to all "ones" state at the start of each frame. The derandomizer can be enabled or disabled through software. The start of randomized data is programmable.

CRC DECODER

The CRC decoder receives serial data from the derandomizer. The CRC decoder computes the CRC remainder from the input data stream using the polynomial is $g(x) = x^{16} + x^{12} + x^5 + 1$. The decoder shift register is initialized to all "ones" at the start of each frame. The CRC decoder compares the computed 16-bit remainder with the error control field in the transfer frame. The CRC decoder can be enabled or disabled through software. The start and end position of the CRC data is programmable.

BIT AND WORD COUNTERS

The bit counter provides an end of word signal (BCTC) to the input control logic. The word size is programmable from 4 to 16 bits/word. The word counter controls the length of the input frame, which is programmable from 2 words to 8192 words. The word counter is a down counter that provides an end-of-frame signal (WCTC) to the input control logic.

INPUT CORRELATOR

The input correlator is an L10C23 digital correlator that performs a bit-by-bit correlation to any sync pattern up to 64 bits in length. All of the bits in the sync pattern can be masked. The sync pattern, mask, and bit error threshold are software programmable using the setup registers on the MONARCH-E. The correlator computes the number of data bits that agree with the sync pattern. The correlation result and its complement are compared with the programmed threshold to determine if a true or inverted sync pattern has been received.

FRAME SYNCHRONIZER

The frame synchronizer control logic uses an adaptive strategy for acquiring frame synchronization which consists of four states: SEARCH, CHECK, LOCK, and FLYWHEEL.

A state diagram of the minor frame synchronizer control logic is shown in **Figure 3-2**. While in the SEARCH state, the board searches the input data stream for an acceptable sync pattern. Once an acceptable pattern is found, the board enters the CHECK or LOCK state depending on the programmed number of check frames. The LOCK state is entered when N consecutive acceptable frames are received, where the number of check frames N is programmable from 0 to 7 frames. If an unacceptable frame is received while in the CHECK state, the board returns to the SEARCH state. The board remains in the LOCK state as long as consecutive acceptable frames are received. If an unacceptable frame is received, the board advances to the FLYWHEEL or SEARCH state depending on the programmed number of flywheel frames. If M consecutive unacceptable frames are received while in the FLYWHEEL state, then the board returns to the SEARCH state. The number of flywheel frames M is programmable from 0 to 7 frames. If an acceptable frame is received while in the FLYWHEEL state, then the frame synchronizer returns to the LOCK state. This method of operation ensures that the board will remain in LOCK even when the received data stream is corrupted by random bit errors.

The frame synchronizer supports a frame sync bit slip window from 0 to ±3 bit periods wide. If a bit slip occurs and a valid sync pattern is received within the programmed window, then the serial input control logic will adjust the bit count to accommodate the bit slip.

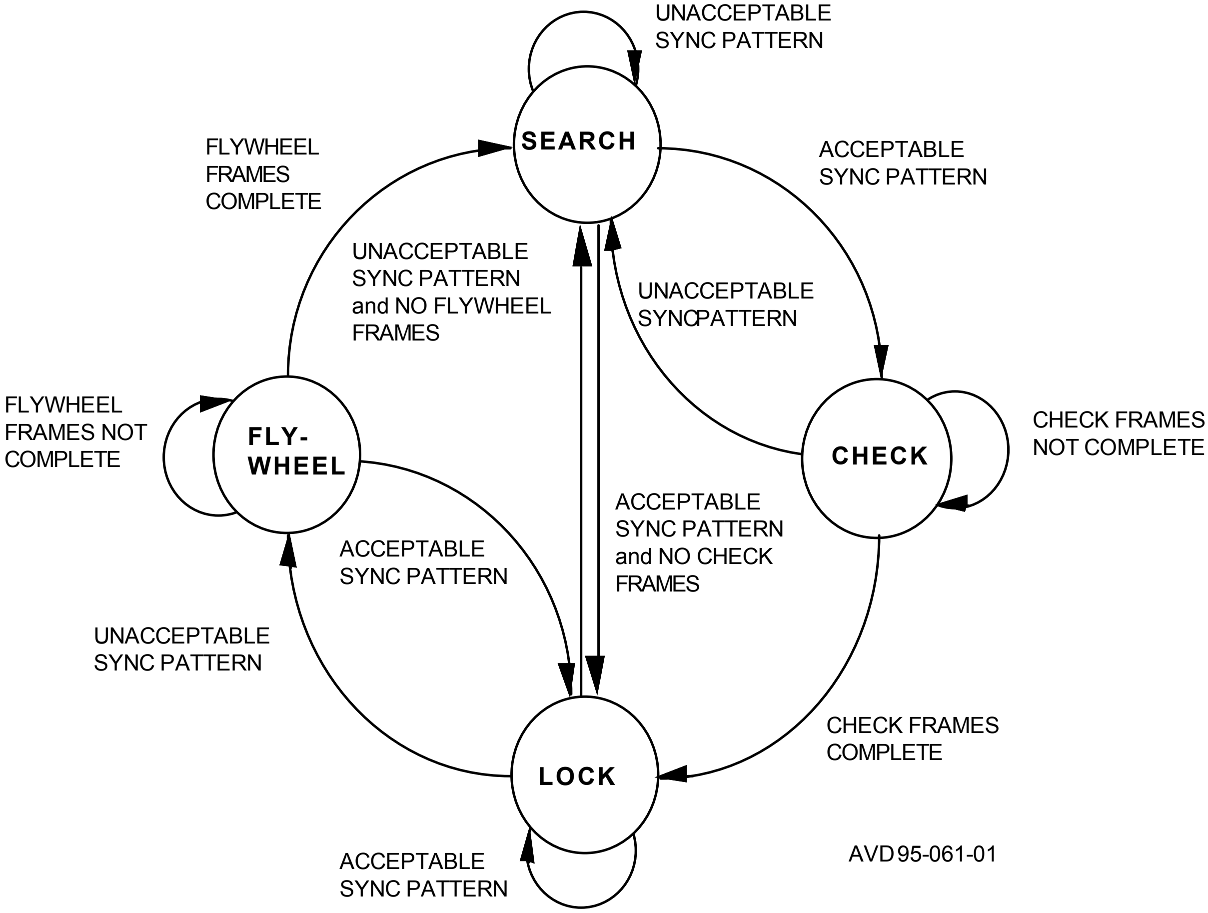


Figure 3-2: Frame Synchronizer State Diagram

The serial input control logic provides autopolarity detection and correction. When this feature is enabled, the control logic searches for both the true and inverted sync pattern. If an inverted sync pattern is detected, then the input data is inverted before it is stored in local memory.

When the correlator is enabled, frame data is discarded until the frame synchronizer is in the LOCK or FLYWHEEL state. The input channel correlator may also be disabled to collect raw data input. When the frame sync is disabled, the sync pattern and mask, search and lock error tolerances, number of check frames, number of flywheel frames, and bit slip window width are not used. However, the frame length and word size must still be defined since they are used by the serial-to-parallel conversion logic.

Serial-to-parallel Shift Register

The serial data from the code converter is sent through the pseudo-derandomizer and CRC decoder before being converted to parallel data. The serial input channel stores either Most Significant Bit (MSB) or Least Significant Bit (LSB) first serial data on the programmed word boundaries using shift registers inside the FPGA. Telemetry words less than or equal to 8 bits in length are stored as bytes and telemetry words greater than 8 bits are stored as 16-bit words.

In CCSDS mode, the parallel data is written into FIFO buffers which feed the Reed-Solomon Decoder. For TDM mode, the parallel data bypasses the Reed-Solomon Decoder and is written directly into the PCI interface FIFO buffers. Because the PCI interface is 32 bits wide, frames are padded to longword (32-bit) boundaries. Maintaining longword alignment improves performance by allowing the PCI DMA controller to perform 32-bit burst transfers to PC memory.

QUALITY ANNOTATION AND TIMESTAMP

The serial input logic can append quality annotation and a 32-bit time stamp to the end of each frame. The quality annotation consists of the selected receiver status register values when the frame was received. The quality annotation includes: frame sync state, number of sync pattern bit errors, autopolarity active flag, bit slip flag, and CRC error flag.

The MONARCH-E provides an internal time tag using a 32-bit tick counter. The 32-bit counter is driven by a clock derived from the PCI bus clock (divided by 4) or an external input signal. The host resets the counter at the beginning of an acquisition phase and records the station time from a PC-based time code processor card. The ground receipt time for each frame is computed from the station time at reset plus the elapsed time indicated by the tick counter value appended to the frame. The MONARCH-E supports time tagging of the leading edge of the first bit or the falling edge of the last bit.

The tick counter method of time tagging gives a very high resolution measurement of the time a frame was received relative to when the 32-bit counter was reset. The error in the time stamp depends on the latency between resetting the 32-bit counter and recording the time from the time code processor card. This latency is the time required for the Host Processor CPU to make back-to-back bus accesses and should be on the order of several microseconds. The tick counter method will not work if the station time is not valid when the board is initialized.

The MONARCH-E board also provides a time tag interrupt (controlled by TMS_IEN) on the receipt of the first bit or last bit in the frame. The host processor can then read the PC-based time code processor card to determine the receipt time. The accuracy of this method depends on the interrupt latency (delay from when the interrupt is asserted until the processor actually reads the time board). The MONARCH-E can generate a TMS interrupt every 16 frames (rather than every frame) when TMS_MFB is set to 1.

MONARCH-E boards with the –T modification also provide a strobe output (TMS_STROBE) that can be connected directly to a time code processor card such as the Apogee ISA-STG2. The time code processor can latch the time when the strobe is asserted until the host processor can read the time. This eliminates the time tag error that can be introduced by interrupt latency. If the time code processor can only hold a single time tag value, then the host processor must read the time before the next time tag interrupt. This may not be possible for small frame lengths (i.e. less than 64 bytes) and high bit rates (> 2 Mbps). The MONARCH-E can generate a TMS interrupt every 16 frames when TMS_MFB is set. The time code processor is used to time tag the first frame in a group of 16 and the MONARCH-E tick counter is used to compute the time tag for the remaining 15 frames in the group.

When the TMS_STROBE is enabled and TMS_MFB (one strobe per frame) is set, then the MONARCH-E tick counter is used to provide a received bit rate measurement. The tick counter value will equal the number of reference clock ticks during the frame. The received bit rate can be computed based on the number of bits in the frame and the reference clock period as shown below:

Measured Bit Rate (bps) = Frame Length (bits) / (Tick Count * Ref Clk Period (sec))

The MONARCH-E time tag logic only works when the Frame Synchronizer is enabled (Receiver RSTR COR_EN = 1). In order to time tag raw serial (throughput) data, the Frame Synchronizer must be enabled (Receiver RSTR COR_EN = 1) and set for Frame Sync Last (Receiver RSTR FSF_L = 0) with the Correlator threshold (Receiver THOLD) set to 0. This will allow the board to capture raw serial data with time tag.

The MONARCH-E board internal pipeline delay depends on the mode of operation. The pipeline delay is the number of bit periods between the time the first (or last bit) arrives at the input to the board and when the time tag is actually latched. For Frame Sync First operation (Receiver RSTR FSF_L = 1), the pipeline delay is 73 bit periods. For Frame Sync Last operation (Receiver RSTR FSF_L = 0), the pipeline delay is 9 bit periods. This pipeline delay also applies for the raw data mode described above. The measured time tag can be corrected to eliminate the pipeline delay as shown below:

Corrected Time Tag = Measured Time Tag - (Pipeline Delay * 1/Measured Bit Rate)

INTERRUPT CONTROL

The interrupt control logic can generate an interrupt request to the Host Processor CPU for the following conditions:

- LOCK- frame synchronizer enters the LOCK state from the SEARCH or CHECK states
- WIN_END_FLG - end of frame is reached
- UNLOCK - frame synchronizer returns to SEARCH state from the LOCK or FLYWHEEL states
- SLIP - a bit slip is encountered
- CRC_ERR - CRC decoder error
- TMS - time stamp strobe asserted
- RSEC - interrupt from the Reed-Solomon Error Correction chip
- OVRN - FIFO overrun

The interrupt request conditions are individually enabled or disabled through software.

Section 3.2
REED-SOLOMON ERROR CORRECTION (RSEC) CHIP

The RSEC chip is an LSI Logic ASIC that was designed by engineers at NASA's Goddard Space Flight Center. The RSEC chip accepts parallel data from the Serial Input Logic FIFO buffer. It performs RS(255,223) VCDU error correction with interleaving depth from 1 to 8, RS(10,6) VCDU header error correction, and real-time quality generation and annotation. Shortened code blocks are also supported. The RSEC chip outputs corrected VCDUs with quality annotation to the PCI Interface. For more information about the RSEC chip, please refer to Reference 2 (521-SPEC-002).

Section 3.3
PCI INTERFACE

The MONARCH-E is a 32-bit, 5V PCI device. The PCI interface is used for control and status as well as high speed data transfer. This interface is implemented by the PLX PCI9080 PCI controller. It provides full PCI compatibility, including support for auto configuration and burst mode operation. The PCI interface transfers frame data to/from the host processor's memory at up to 132 MB/sec. Additionally, the MONARCH-E can be programmed to pass board interrupts to the PCI bus.

The first two base address registers in the PCI configuration header, PCIBAR0 and PCIBAR1, provide PCI memory and I/O access to the PLX shared registers respectively. All of these registers can be read and written as either memory or I/O. The base address register regions each occupy 128 bytes of PCI address space. See the PLX PCI9080 data sheet (Reference 3) for a full description of these registers. The third base address register, PCIBAR2 (Local Address Space 0), maps to the logic control registers for the Reed-Solomon Decoder, the serial input, the Reed-Solomon Encoder, and the serial output. This space requires 8 KB of PCI memory space, and is used to access the MONARCH-E registers at the offsets listed below.

Reed-Solomon Error Correction chip	0x0000
Receiver FPGA	0x0400
Reed-Solomon Encoder FPGA	0x0800
Transmitter FPGA	0x0C00
Receiver/Transmitter Xilinx Configuration	0x1000

See Chapter 4, **Programming Interface,** for a full description of the registers in this region.

Section 3.4
REED SOLOMON ENCODER

The Reed-Solomon Encoder is implemented in a Field Programmable Gate Array. The encoder accepts parallel data from the PCI interface and performs RS(255,223) VCDU encoding with an interleave depth from 1 to 8. Shortened code blocks are also supported. Reed-Solomon parity data can be appended to the data stream or overwrite the host supplied data. The Reed-Solomon Encoder outputs interleaved code words to the Serial Output Logic. From a hardware perspective, the Reed-Solomon Encoder includes a parallel-to-serial converter, a FIFO, and bit and word counters, as shown in **Figure 3-3**.

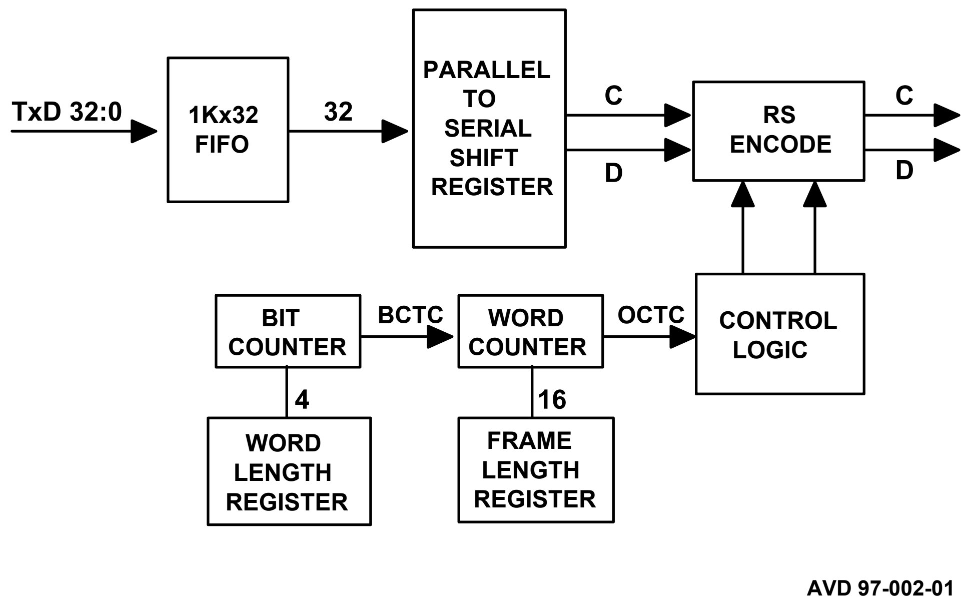


Figure 3-3: Reed Solomon Encoder Logic

Section 3.5
SERIAL OUTPUT LOGIC

The MONARCH-E output channel reads frame data from PC memory via the PCI bus and outputs serial data and clock at up to 25 Mbps. The output logic performs CRC encoding, pseudo-randomization, and convolutional encoding.

The MONARCH-E supports both TDM and CCSDS telemetry formats. For CCSDS output, the PCI interface transfers frame data to the Reed-Solomon Encoder which then transfers coded frames to the serial output logic. For TDM output, the PCI interface transfers frame data directly to the serial output logic, bypassing the Reed-Solomon Encoder.

A block diagram of the MONARCH-E serial output logic is shown in **Figure 3-4**. The main blocks are the convolutional encoder, the code converter, the pseudo-randomizer, the CRC encoder, the word counters, the serial output control logic, and the interrupt request logic. All of the serial output logic except the FIFO buffers are implemented in a single field programmable gate array. From a hardware perspective, the parallel-to-serial shift registers, FIFO buffers, and bit counters are contained in the Reed-Solomon Encoder chip.

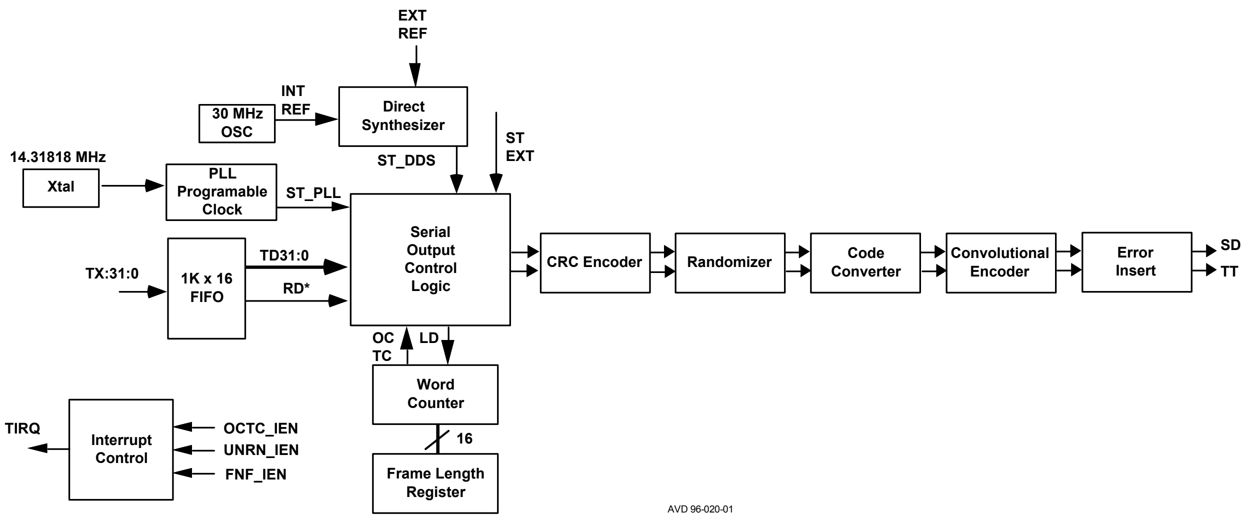


Figure 3-4: MONARCH-E Serial Output Logic

CRC ENCODER

The CRC encoder receives serial data from the parallel-to-serial shift register. The CRC encoder computes the CRC remainder from the input data stream and overlays the 16-bit result at the end of the frame. The generator polynomial is $g(x) = x^{16} + x^{12} + x^5 + 1$. The encoder shift register is initialized to all "ones" at the start of each frame. The CRC encoder can be enabled or disabled through software. The start and end positions are programmable to allow the sync markers and/or Reed-Solomon parity to be excluded.

RANDOMIZER

The method for ensuring sufficient transitions in the output stream is to exclusive-OR each bit of the Code Block or Transfer Frame with a standard pseudo-random sequence. The pseudo-randomizer receives serial data from the CRC encoder and outputs the randomized serial stream to the code converter. The sequence is generated using the polynomial $h(x) = x^8 + x^7 + x^5 + x^3 + 1$. The sequence generator is re-initialized to all-ones state at the start of each frame. The derandomizer can be enabled or disabled through software. The start position is programmable to allow the sync marker to be excluded.

CODE CONVERTER

The code converter accepts NRZ-L input from the pseudo-randomizer and outputs NRZ-L, S, or M or Biphase-L, S, or M codes. The code conversion mode is software selectable. Code conversion can be performed before and after convolutional encoding.

CONVOLUTIONAL ENCODER

The convolutional encoder accepts serial data from the code converter and outputs encoded serial data. The basic code for cross-support is a rate 1/2, constraint-length 7, transparent convolutional code. The connection vectors for the convolutional code are G1 = 1111001 and G2 = 1011011 where G1 is the first symbol and symbol inversion is applied to G2. The CCSDS recommendation encourages all agencies to adopt the convention described here which is the NASA-GSFC convention. However, the parity order and parity inversion are programmable for the MONARCH-E convolutional encoder to support applications which use different ordering or inversion of the two parity checks. The convolutional encoder can be enabled or disabled through software.

BIT AND WORD COUNTERS

The bit counter provides an end of word signal (BCTC) to the output control logic. The word size is programmable from 4 to 16 bits/word. The word counter is a 16 bit down counter that provides an end of block signal (WCTC) to the output control logic. The block length is programmable from 2 to 64 K words. The word counter is used to accommodate the 32-bit frame alignment by skipping pad bytes.

SERIAL OUTPUT CONTROL LOGIC

The serial output control logic controls the flow of data from the FIFO buffers to the parallel-to-serial shift register, as shown in **Figure 3-5**. Data output is controlled by the START and STOP commands and the FIFO buffer not empty or half full flags. When a START command is given and data is present in the FIFO buffer, serial output begins. Serial output halts if and when the FIFO buffer becomes empty and resumes when data is available again. When a STOP command is given, serial output halts and the control logic returns to the idle state.

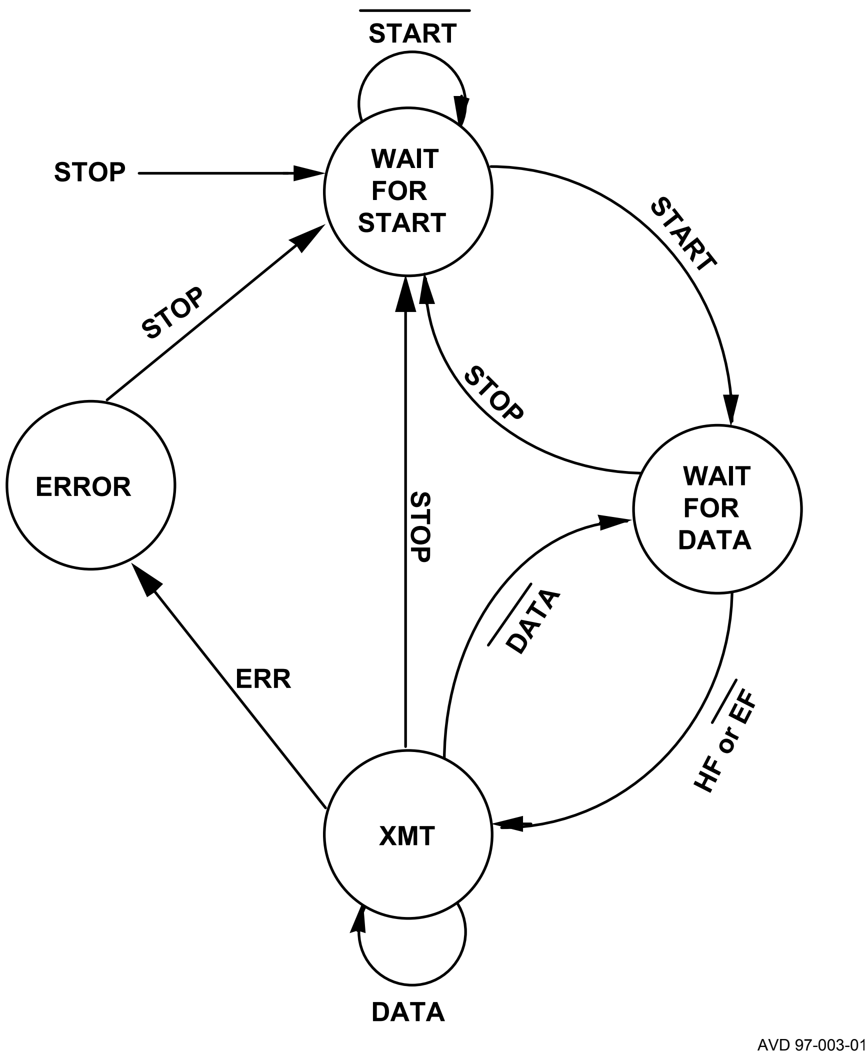


Figure 3-5: Serial Output Control State Diagram

INTERRUPT CONTROL

The interrupt control logic generates an interrupt request to the Host Processor CPU based on the state of the output channel. The interrupt conditions are:

- EOF - end of frame
- FAF - FIFO buffer almost full
- LOD - loss of data (i.e. underrun on frame boundary)
- UNRN - underrun not on frame boundary

The interrupt request conditions are individually enabled and disabled through software.

Section 3.6
FREQUENCY SYNTHESIZERS

The MONARCH-E contains two transmit clock frequency synthesizers: a direct digital synthesizer (DDS) and a PLL based programmable clock chip. The DDS provides very high resolution frequency output from 10 Hz to 10 MHz. The programmable clock chip provides a lower resolution frequency output from 320 kHz to 100 MHz.

The serial output clock is selected through software to be either the onboard DDS output (ST_DDS), the onboard PLL clock chip (ST_PLL), or an externally generated send timing signal (ST_TTL or ST_422).

The DDS output frequency is set by the programmed 32-bit phase offset. The DDS reference signal is an onboard 30 MHz oscillator that has a long term stability of ±25 ppm. The DDS reference can also be phase-locked to an external 5 MHz reference.

The frequency of the Direct Digital Synthesizer (DDS) is set by the phase increment value. The frequency control word is loaded serially by writing to the DDSCLK Register. After the control word is loaded, writing to the DDS_EN bit of the Transmitter Control Register transfers the frequency control word to the phase accumulator.

The DDS control word is programmed by the following formula:

$N = 2^{32}$ (FREQ/OSC) where

N = DDS control word

FREQ = desired frequency (MHz)

OSC = frequency of reference oscillator (30 MHz)

The 8-bit digital output of the DDS is then fed into a digital-to-analog converter (DAC) to get an analog sine wave output. The sine wave output is then passed through an elliptical low pass filter with a 10 MHz frequency cutoff. The smoothed sine wave is finally fed into a video comparator to generate a TTL clock signal. The transmit clock provided by the direct digital synthesizer (DDS) has 5 digits of resolution over the 10 MHz frequency range.

The programmable clock generator, ICD2051, can output any frequency between 320 kHz and 100 MHz. The phase-locked loop oscillator input is derived from a 14.31818 MHz crystal. The ICD2051 output frequency is set by a 22-bit control word**.** The ICD2051 normally produces an output frequency within 0.1% of the desired frequency. The resolution is calculated to be about 3%.

The equation to calculate the oscillator control word is as follows:

FREQ = 2 * REF * (P/Q) where

P, Q = part of a 22-bit control word used to program the clock, where P and Q are a 7-bit binary value set

REF = frequency of reference oscillator (14.31818 MHz)

See References 4 and 5 for more details on programming these clocks.

Chapter 4
Programming Interface

Section 4.1
OVERVIEW

The host computer communicates with the board via control and status registers in the PLX PCI9080 Interface, the Reed-Solomon Error Correction (RSEC) chip, and the Xilinx FPGAs. A full description of the register interface for the first two devices can be found in their respective data sheets (References 2 and 3). The following sections define the PLX PCI9080 configuration, describe how to configure the Xilinx devices, and provide detailed descriptions of the receiver and transmitter control and status registers.

Section 4.2
PLX PCI9080 PCI INTERFACE CONFIGURATION

The PLX PCI9080 supports the PCI Configuration Registers as defined in the PCI Local Bus Specification. The settings for the PCI Configuration Registers are loaded from an onboard EEPROM at power up. The MONARCH currently uses the PLX Vendor ID (10B5 hex) and Device ID (9060 hex). Future versions of the board will use Avtec's PCI SIG assigned Vendor ID (156A hex).

The MONARCH-E resources are accessed from the PCI bus through the PLX PCI9080. The PLX PCI9080 provides four PCI Base Address registers in the PCI Configuration space as outlined in **Table 4-1** below. The MONARCH-E only uses PCIBAR0 and PCIBAR2 (Local Address Space 0).

Table 4-1: PLX PCI9080 PCI Base Address Register Utilization

PCI Base Address Register (PCIBAR)	Function
0	Memory mapped access to the PCI9080 control/status registers
1	I/O mapped access to the PCI9080 control/status registers (not used by MONARCH-E)
2	PCI9080 Local Address Space 0
3	PCI9080 Local Address Space 1 (not used by MONARCH-E)

The host queries the PCI Base Address Registers as defined in the PCI Local Bus specification to determine the size of each of these address spaces. The PCI9080 control and status registers

(PCIBAR0) require 256 bytes of address space. The Local Address Space 0 (PCIBAR2) requires 8 KB (0x2000) of address space.

The PLX PCI9080 Local Address Space 0 Region Descriptor Register (LBRD0) at offset 18 (hex) from PCIBAR0 must be set to 0x400F030F before performing any access to Local Address Space 0.

The MONARCH-E control and status registers are contained in the PCI9080 Local Address Space 0 as outlined below.

Table 4-2: MONARCH-E Register Address Map

PLX Local Address Space 0 (PCIBAR2) Offset (Hex)	Function
0000	Reed-Solomon Error Correction (RSEC) Chip control/status registers
0400	Receiver FPGA Registers
0800	Reed-Solomon Encoder FPGA Registers
0C00	Transmitter FPGA Registers
1000	Receiver/Transmitter Xilinx Configuration

Section 4.3
XILINX CONFIGURATION

There are three Xilinx FPGAs on the MONARCH-E which must be configured each time the board is powered up. They can also be reconfigured by following the same steps if a different logic configuration is required.

The Serial Input (Receiver) and Serial Output (Transmitter) devices must be configured first. The PLX PCI9080 General Purpose Output (`USERo`) signal is used to reset these devices. The `USERo` signal is controlled by bit 16 of the PLX PCI9080 CNTRL Register (PCIBAR0 + 0x6C). It should be pulsed high for at least 100 ms to initialize the devices and begin the configuration process. The configuration file can then be written to the Xilinx Configuration Base Address (PCIBAR2 + 0x1000) one byte at a time. To flush the data through the internal pipeline an additional five bytes must be written. The value of these bytes is not important. Upon completion the configuration LEDs (LED1 and LED2) will light up and the `USERi` signal on the PLX PCI9080 will be driven high. The `USERi` signal state is reflected in bit 17 of the PLX PCI9080 CNTRL Register (PCIBAR0 + 0x6C).

The steps on configuring the Receiver and Transmitter FPGAs are as follows:

1) Set the PLX PCI9080 `USERo` to 1.
2) Wait 100 msec.
3) Set the PLX PCI9080 `USERo` to 0.

4) Wait 1 msec.
5) Write the FPGA binary one byte at a time to offset 0x1000 in Local Address Space 0.
6) After downloading the FPGA binary, write five dummy bytes (0) to offset 0x1000 in Local Address Space 0. Note: the LEDs illuminate after two dummy bytes, but `USERi` is not set until after the fourth byte.
7) Verify that the FPGAs are configured by reading the PLX PCI9080 `USERi` bit (1 = configured, 0 = not configured).

After the Serial Input and Serial Output devices are configured, the Reed-Solomon Encoder device can be configured. Since the Reed-Solomon Encoder FPGA is configured serially through the Transmitter FPGA, the Receiver and Transmitter FPGAs must be configured first. The configuration process begins by resetting the Reed-Solomon Encoder device. This is accomplished by setting the XCENC_PROG bit in the Transmitter Control Register (PCIBAR2 + 0xC08). The board will acknowledge the reset by driving the XCENC_INIT* bit in the Transmitter Status Register (PCIBAR2 + 0xC00) low. The XCENC_PROG bit should be cleared to complete the reset cycle. The XCENC_INIT* bit will then be driven high to indicate the device ready to accept configuration data. The data must be written to the Encoder Configuration Register (PCIBAR2 + 0xC2C) serially beginning with the LSB of the first byte. If an error occurs while configuring the device, the XCENC_INIT* bit will be driven low and the process must be restarted. Upon successful completion the final configuration LED (LED3) will become illuminated and the XCENC_DONE bit in the Transmitter Status Register will be driven high.

The steps on configuring the Reed-Solomon FPGA are as follows:

1) Set bit XCENC_PROG in the Transmitter Control Register (offset 0x0C08 in Local Address Space 0).
2) Wait for XCENC_INIT to go low.
3) Clear bit XCENC_PROG.
4) Wait for XCENC_INIT in the Transmitter Status Register TSTAT0 to go high (approximately 2 ms).
5) Download the FPGA binary serially (LSB first) by writing to offset 0x0C2C (XCENC) in Local Address Space 0. Note: the LED illuminates and XCENC_DONE goes high immediately after the last bit is written.
6) After downloading the FPGA binary, verify that the FPGA is configured by reading the XCENC_DONE bit in the Transmitter Status Register TSTAT0 (offset 0x0C00 in Local Address Space 0). A one indicates that the Reed-Solomon Encoder is configured.

Section 4.4
PCI DATA TRANSFER

The MONARCH-E uses the PCI9080 Direct Memory Access (DMA) channels to transfer data to/from system memory. DMA channel 0 is used to transfer data from the input FIFO to memory and DMA channel 1 is used to transfer data from memory to the output FIFO. Both channels operate in Demand Mode. The receive DMA request (DREQ0) is generated when received data is available in the input FIFO. The transmit DMA request (DREQ1) is generated when there is space in the output FIFO.

A brief overview of the DMA channel control and status registers is provided in the table below. For more information, please refer to the PLX PCI9080 data sheet.

Table 4-3: PLX PCI9080 DMA Control and Status Registers

Register Name	Comment
DMA Mode (DMAMODEx)*	Initialize to 0x000019C3 - 32-bit bus width, no wait states, READYi enabled, BTERM enabled, Burst enabled, no chaining, local address constant, demand mode. Bit 9 can be set to enable DMA chaining. Bit 10 Done Interrupt is set to enable DMA Done Interrupts.
DMA PCI Address (DMAPADRx)	PCI address of system memory for transfer.
DMA Local Address (DMALADRx)	Local address of FIFOs. Should be set to 0x80000000.
DMA Transfer Size (DMASIZx)	Number of bytes to transfer.
DMA Descriptor Pointer (DMADPRx)	Should be set to 0x8 for DMA Channel 0 and 0x0 for DMA channel 1.
DMA Command/ Status (DMACSRx)	Should be set to 0x3 to enable and start a DMA transfer.

* x = channel 0 or channel 1

The DMA Arbitration Register (DMAARB) and the DMA Threshold Register (DMATHR) should be left at their default values.

All PCI Data transfers to/from the MONARCH-E are long-word aligned to maximize the PCI transfer rate. The MONARCH-E receiver will insert pad bytes when receiving frames that are not evenly divisible by 4 bytes as shown in the figure below.

Frame 1 (Number of bytes = Frame Length)	Pad	Frame 2 (Number of bytes = Frame Length)	Pad

Figure 4-1: Received Frame Format with Pad and No Annotation

The number of pad bytes is given by the formula below:

Number of Pad bytes = 4 – Frame Length % 4

When using the Reed-Solomon Decoder (RCR Bypass = 0), the Reed-Solomon Decoder frame length should be programmed with the length of the Frame Data plus Pad plus Time Tag and Frame Sync status.

The MONARCH-E transmitter expects frames to be DMA transferred in the same format as shown in **Figure 4-1**. The MONARCH-E transmitter will discard the pad bytes before transmitting the data. When the Reed-Solomon encoder is enabled in overwrite (RSCR2 RS_OVWR = 1), then PCI transfer data length should include space for the Reed-Solomon check symbols and equal the entire Transfer Frame length plus Pad. When the Reed-Solomon encoder is enabled in append mode (RSCR2 RS_OVWR = 0), then the PCI transfer data length should equal the Transfer Frame length minus the check symbol length (32 * Interleave Depth) plus Pad.

Section 4.5
APPENDED STATUS FORMAT

The Serial Input Logic and the Reed-Solomon Decoder can be configured to append frame specific status to each frame to allow the data to be processed after it is collected. The Serial Input Logic can append a 32-bit time tag and 16-bit Frame Sync status to each frame. The Reed-Solomon Decoder can append 32-bytes of quality annotation to each frame.

The following figures illustrate the format of the frame data with each type of appended status. The 'Pad' bytes insure that the frame is long word aligned before the time tag, frame sync status, and Reed-Solomon status are appended.

Frame Data (Number of bytes = Frame Length)	Pad

Figure 4-2: Frame Format with No Annotation

Frame Data (Number of bytes = Frame Length)	Pad	Time Tag & Frame Sync QA (16 bytes)

Figure 4-3: Frame Format with Frame Sync Status and Time Tag

Frame Data (Number of bytes = Frame Length)	Pad	Time Tag & Frame Sync QA (16 bytes)	Reed-Solomon QA (32 bytes)

Figure 4-4: Frame Format with Frame Sync Status, Time Tag, and Reed-Solomon Status

When using the Reed-Solomon Decoder (RCR Bypass = 0), the Reed-Solomon Decoder frame length should be programmed with the length of the Frame Data plus Pad plus Time Tag and Frame Sync status.

The tables below provide the byte and bit-level definitions for each of the appended status areas. **Table 4-4** illustrates the format of the MONARCH-E time tag and frame sync status. All of the quality annotation fields are ‘Big-Endian’ so the most significant bytes will appear in memory before (at a lower address) than the least significant bytes. **Table 4-5** provides the frame sync status bit field definitions. **Table 4-6** describes the 32-byte Reed-Solomon Decoder Quality Annotation. **Table 4-7** supplements the previous table and provides the details of the Reed-Solomon Decoder Terse Quality field. More detailed information on the Reed-Solomon Quality Annotation can be found in Reference [2].

Table 4-4: MONARCH-E Time Tag and Frame Sync Appended Status Format

Parameter	Size	Description
Application Flags	8 bytes	8 bytes of Pad are inserted between the long-word aligned frame data and the Time Tag and Frame Sync Status.
Time Tag	4 bytes	32-bit Time Tag tick counter. This tick counter will equal the number of reference clock ticks since the tick counter was reset (TMS_STROBE = 0 or TMS_STROBE =1 and TMS_MFB = 1) or it will represent the number of ticks that occurred during the current frame (TMS_STROBE = 1 and TMS_MFB = 0).
Frame Sync Status High	2 bytes	16-bit Frame Sync status field. Bit values are defined in **Table 4-5**.
Frame Sync Status Low	2 bytes	Undefined.

Table 4-5: Frame Sync Appended Status Register

Bit	Name	Description
15:14	SICQ <1:0>	This 2-bit field indicates the state of the serial input channel control logic as shown in the table below: SICQ<1:0> — Input Channel State 00 — SEARCH 01 — CHECK 10 — FLYWHEEL 11 — LOCK The status appended to a particular frame is the state AFTER the frame was received. If the frame is the last frame received prior to a drop out (return to SEARCH), then the state for that frame will be SEARCH.
13	APC_FLG	Autopolarity Correct Flag. This bit is set by the serial input control logic locking to an inverted sync pattern.
12	CRC_ERR_FLG	CRC Error Flag. This bit is set if the value calculated by the CRC Decoder does not match the CRC field in the data.
11	SLIP_FLG	Slip Flag. This bit indicates that a bit slip has been detected.
10:8	SDIS <2:0>	Slip Distance. This field indicates the magnitude and direction of the previous frame’s slip window. The distance in bit periods is given by SDIS <2:0> - SLIPSEL <2:0>. For example, if the allowable slip window (SLIPSEL) is 1 and SDIS =1, no slip occurred. If SLIPSEL=1 and SDIS=0, the sync pattern occurred one bit later than expected.

Bit	Name	Description
7	TMSV_FLG	Time Stamp Overflow Flag. This bit indicates that the Time Stamp Counter has overflowed.
6	UNLOCK_FLG	Unlock Flag. This bit indicates that the serial input channel has entered the SEARCH state from a LOCK or FLYWHEEL state (dropped frame lock).
5	LOCK_FLG	Lock Flag. This bit indicates that the logic has entered the LOCK state from a SEARCH or CHECK state.
4:0	X	Undefined.

Table 4-6: 32-byte Reed-Solomon Decoder Chip Annotation Format

Parameter	Size	Description
User Annotation [1]	4 bytes	32-bit user defined annotation from Serial Input Configure Extended Options
User Annotation [2]	4 bytes	32-bit user defined annotation from Serial Input Configure Extended Options
Frame Counter	4 bytes	32-bit count which is incremented for each frame passed through the RS decoder chip. Starting value is zero on enable.
Routing Data	1 byte	Not Used.
Terse Quality	1 byte	Contains high level information about the current frame. Defined in the table shown below.
Errors Corrected	1 byte	Number of RS (255,223) errors corrected
Header Errors Corrected	1 byte	Number of RS (10,6) header errors corrected
Errored Codewords	1 byte	Number of codewords with errors in the current frame. An uncorrectable codeword is also errored, and this is counted in both fields.
Uncorrectable Codewords	1 byte	Number of uncorrectable codewords in the current frame.
Error Bitfield	2 bytes	16-bit field that indicates which codewords were errored. Bit 0 refers to the first codeword, bit 1 refers to the second, etc. As with the previous quality annotation fields, if the uncorrectable bit is set for a certain codeword, the same bit is set in the errored bit field.
Uncorrectable Bitfield	2 bytes	16-bit field that indicates which codewords were uncorrectable. Bit 0 refers to the first codeword, bit 1 refers to the second, etc.
Bit Errors	2 bytes	Total number of bit errors found and corrected by the RS (255,223) decoder. The block decoder can correct up to 16 symbol errors per codeword. Because each symbol is a byte, the decoder could possibly correct 128 bits in a single codeword. This field reports the total number of bits corrected by the block decoder.
Errors Per Codeword	8 bytes	16-nibble fields; each nibble indicates the total number of errors per codeword. The first nibble refers to codeword 16, and the last nibble refers to codeword 1.

Table 4-7: Reed-Solomon Decoder Chip Terse Quality Definition (supplement to previous table)

Bit #	Description
7	Long frame
6	Short frame

Bit #	Description
5	Unrouteable frame
4	Uncorrectable frame
3	RS (255,223) uncorrectable frame
2	RS (10,6) uncorrectable header
1	RS (255,223) found errors in frame
0	RS (10, 6) found errors in header

Section 4.6 PCI INTERRUPTS

The PLX Interrupt Control/Status Register (INTCSR) is used to control the generation of PCI interrupts by the MONARCH-E. The Interrupt Control/Status Register is located at offset 0x68 from PCIBAR0 in the PCI9080 register map. The following table outlines the INTCSR bits that are pertinent to the MONARCH-E.

Table 4-8: PLX PCI9080 Interrupt Control/Status Register

PLX Interrupt Control/Status Register Bit	Function
8	PCI Interrupt Enable. Value of 1 enables PCI interrupts.
11	PCI Local Interrupt Enable. Value of 1 enables a local interrupt input to generate a PCI interrupt. The local interrupt input is driven by the FPGA interrupt logic. (Please see the receiver and transmitter programming sections for more detailed information).
15	Local Interrupt Status. Value of 1 indicates the local interrupt from the FPGA is active.
18	Local DMA Channel 0 Interrupt Enable. Value of 1 enables DMA Channel 0 interrupts.
19	Local DMA Channel 1 Interrupt Enable. Value of 1 enables DMA Channel 1 interrupts.
21	DMA 0 Interrupt Active. Value of 1 indicates the DMA Channel 0 interrupt is active. Clearing the DMA status bit also clears the interrupt.
22	DMA 1 Interrupt Active. Value of 1 indicates the DMA Channel 1 interrupt is active. Clearing the DMA status bit also clears the interrupt.

The receive DMA channel 0 can be programmed to generate an interrupt when a frame of data has been received and transferred into memory. The transmit DMA channel 1 can be programmed to generate an interrupt when a frame of data has been transferred from memory to the output FIFO. The DMA channel 1 interrupt does not indicate that the board has actually transmitted the data. The Transmitter End of Frame (EOF) interrupt can be used to notify the system processor when a frame of data has actually been transferred.

Section 4.7
RECEIVER REGISTERS

The tables below list the different programmable registers for the Receiver. All accesses to the Xilinx registers must be 32 bits wide, although not all of the bits may be significant. To ensure compatibility with future enhancements, all reserved bits should be set to 0 when writing to a register and ignored when reading from a register.

Table 4-9: Receiver Register Map – Write Only

Offset	Register	Name
0	RCOM	Receiver Command Register
4	RICR	Receiver Interrupt Control Register
8	RCR	Receiver Control Register
C	RSTR	Receiver Strategy Register
10	RFL	Receiver Frame Length
14	SYPR	Sync Pattern Register
18	MASK	Sync Mask Register
1C	THOLD	Threshold Register
20	RCR2	Receiver Control Register 2
24	RX_RDLEN	Receiver Randomizer Starting Offset
28	RX_CDLEN	Receiver CRC Starting Offset
2C	RX_DEND	Receiver CRC Ending Offset
30		Reserved
34		Reserved
38		Reserved
3C	RGRST	Receiver Global Reset

Table 4-10: Receiver Register Map – Read Only

Offset	Register	Name
0	RSTAT0	Receiver Status Register 0
4	RSTAT1	Receiver Status Register 1

RECEIVER COMMAND REGISTER (RCOM)

Address: REG + 0x0400

Type: Write Only

Register Descriptions:

The Receiver Command Register (RCOM) is a 16-bit register. The table below defines the bit fields.

Table 4-11: Receiver Command Register (RCOM)

Bit	Name	Description
15		Reserved.
14	TMS_RST	Time Stamp Reset. Writing a 1 to this bit resets the Time Stamp.
13	CLK_ACK	Clock Acknowledge. Writing a 1 to this bit clears the CLK_FLG in the Receiver Status Register 0.
12:10		Reserved.
9	TMS_ACK	Time Stamp Acknowledge. Writing a 1 to this bit clears the TMS_FLG in the Receiver Status Register 1.
8	CRC_ERR_ACK	CRC Error Acknowledge. Writing a 1 to this bit clears the CRC_ERR_FLG in the Receiver Status Register 0.
7	PCIOVRN_ACK	PCI FIFO Overrun Acknowledge. Writing a 1 to this bit clears the PCIOVRN_FLG in the Receiver Status Register 1.
6	UNLOCK_ACK	Unlock Acknowledge. Writing a 1 to this bit clears the UNLOCK_FLG in the Receiver Status Register 0.
5	WIN_END_ACK	Window End Acknowledge. Writing a 1 to this bit clears the WIN_END_FLG in the Receiver Status Register 0.
4	SLIP_ACK	Slip Acknowledge. Writing a 1 to this bit clears the SLIP_FLG in the Receiver Status Register 0.
3	RSDOVRN_ACK	Reed-Solomon Decoder Overrun Acknowledge. Writing a 1 to this bit clears the RSDOVRN_FLG in the Receiver Status Register 1.
2	LOCK_ACK	Lock Acknowledge. Writing a 1 to this bit clears the LOCK_FLG in the Receiver Status Register 0.
1	STOP	Writing a 1 to this bit issues a STOP command to the serial input control logic. The input control logic will stop storing data in the input FIFO buffer at the end of the current input word.
0	START	Writing a 1 to this bit issues a START command to the serial input control logic. In pattern recognition mode, the input correlator will begin to search for a valid sync pattern. In raw data mode, the input channel will begin storing data in the input FIFO buffer.

RECEIVER INTERRUPT CONTROL REGISTER (RICR)

Address: REG + 0x0404

Type: Write Only

Register Descriptions:

The Receiver Interrupt Control Register (RICR) is a 16-bit register. The table below defines the bit fields.

Table 4-12: Receiver Interrupt Control Register (RICR)

Bit	Name	Description
15:10		Reserved.
9	TMS_IEN	Time Stamp Interrupt Enable. This bit enables and disables interrupt requests on
8	CRCERR_IEN	CRC Error Interrupt Enable. If this bit is set and the decoder is enabled, an interrupt request occurs when a CRC bit error is detected.
7	PCIOVRN_IEN	PCI FIFO Overrun Interrupt Enable. This bit enables and disables interrupt requests on PCI FIFO overruns. An overrun occurs whenever the serial input channel tries to write data into the FIFO (which feeds the PCI bus) when it is full.
6	UNLOCK_IEN	Unlock Interrupt Enable. This bit enables and disables interrupt requests on entering the SEARCH state from either the LOCK or FLYWHEEL state.
5	WIN_END_IEN	Window End Interrupt Enable. If this bit is set, an interrupt request occurs when the serial input control reaches the end of frame.
4	SLIP_IEN	Slip Interrupt Enable. This bit enables and disables interrupt requests on bit slip detection.
3	RSDOVRN_IEN	Reed-Solomon Decoder Overrun Interrupt Enable. This bit enables and disables interrupt requests on Reed-Solomon Decoder overruns. An overrun occurs whenever the serial input channel tries to write data into the FIFO (which feeds the Reed-Solomon Decoder) when it is full.
2	LOCK_IEN	Lock Interrupt Enable. This bit enables and disables interrupt requests on entering the LOCK state from either the SEARCH or CHECK state.
1:0		Reserved.

RECEIVER CONTROL REGISTER (RCR)

Address: REG + 0x0408

Type: Write Only

Register Descriptions:

The Receiver Control Register (RCR) is a 16-bit register. The table below defines the bit fields.

Table 4-13: Receiver Control Register (RCR)

Bit	Name	Description
15:14	SRC_SEL <1:0>	Data Source. A 0 selects the TTL receive data input, a 1 selects the RS-422 receive data input, 2 is reserved, and a 3 selects internal loop back.
13	RD_INV	This bit selects the polarity of the received data input. For RS-422, a 1 selects inverted data, and a 0 selects true data. For TTL, a 0 selects inverted data, and a 1 selects true data.
12	RC_INV	This bit selects the polarity of the received clock input. For RS-422, a 1 inverts the incoming clock's polarity, and a 0 leaves the incoming clock's polarity unchanged. For TTL, a 0 inverts the incoming clock's polarity, and a 1 leaves the incoming clock's polarity unchanged.
11	AUX_INV	This bit selects the polarity of the Auxiliary Signal. A 1 selects inverted data, and a 0 selects true data.
10:8	ICSEL <2:0>	Input Code Select. This 3-bit field identifies which of the different input codes, as shown in the table below, is used. ICSEL <2:0> — Input Codes 000 — NRZ-L 001 — NRZ-S 010 — NRZ-M 011, 111 — Reserved 100 — Biphase-L 101 — Biphase-S 110 — Biphase-M
7	BYPASS	This bit selects whether the Reed-Solomon Error Correction chip is bypassed. A 1 bypasses the chip, and a 0 includes the chip.
6:5		Reserved.
4	DERAND_MODE	This bit selects whether the Derandomizer Offset is used. A 1 disables the offset. A 0 selects the offset.
3	DERAND_EN	A 1 enables the Derandomizer. A 0 disables the Derandomizer.
2	CRC_MODE	This bit selects whether the CRC offset is used. A 1 disables the offset. A 0 selects the offset.
1	CRC_EN	A 1 enables the CRC verification. A 0 disables the CRC verification.
0	LSB_MSB	Selects the order in which bits are written into the serial-to-parallel converter. 0 is MSB; 1 is LSB.

RECEIVER STRATEGY REGISTER (RSTR)

Address: REG + 0x040C

Type: Write Only

Register Descriptions:

The Receiver Strategy Register (RSTR) is a 16-bit register. The table below defines the bit fields.

Table 4-14: Receiver Strategy Register (RSTR)

Bit	Name	Description
15		Reserved.
14	COR_EN	Correlator Enable. This bit selects pattern recognition mode (COR_EN=1) or raw data mode (COR_EN=0).
13	FSF_L	Frame Sync First or Last. A 1 selects frame sync first operation, and a 0 selects frame sync last operation.
12	APC_EN	Autopolarity Correct Enable. If this bit is set, the serial input control logic will lock to a true or inverted sync pattern and invert the data if an inverted sync pattern is received. Otherwise, the serial input control logic will lock only to a true sync pattern. If the correlator is disabled (COR_EN=0), the bit has no effect.
11:8	INV_ERR <3:0>	Inverted Error Threshold. This field is used in Autopolarity detection. The inverted error threshold for the input correlator is set by writing the appropriate 4-bit value (0 to 15) to the inverted error threshold register. Whenever the total number of bits in disagreement are equal to or greater than the number in the threshold register, the inverted sync flag will go high.
7:6	SLIPSEL <1:0>	The Slip Select sets the number of allowable slipped bits per frame. Whenever the total number of slipped bits is less than the number in the SLIPSEL register, the Slip Flag will go high. The logic will account for the slip and continue normally.
5:3	FFR <2:0>	Flywheel Frame Count. This register sets the number of incorrect sequential sync patterns (0 to 7) required before lock is lost.
2:0	CFR <2:0>	Check Frame Count. This register sets the number of sequential sync patterns (0 to 7) required before lock is achieved.

RECEIVER FLAME LENGTH REGISTER (RFL)

Address: REG + 0x0410

Type: Write Only

Register Descriptions:

The Receiver Frame Length Register (RFL) is a 16-bit register. This register indicates the number of words in each frame. It should be programmed to the desired frame length minus one. The 16-bit value is loaded into the octet counter at the start of each frame. Bit 15 is the MSB and bit 0 is the LSB.

SYNC PATTERN REGISTER (SYPR)

Address: REG + 0x0414

Type: Write Only

Register Descriptions:

The Sync Pattern Register (SYPR) is a 16-bit serially programmable register. The 64-bit sync pattern for the input correlator is programmed serially by performing 64 writes to the Sync Pattern Register.

For sync patterns less than 64 bits and frame sync first operation, the desired sync pattern is shifted into the correlator followed by masked fill bits. For frame sync last operation, the fill bits (64 minus sync pattern length) are shifted into the correlator first followed by the desired sync pattern.

Bits are written out MSB first. Write 00000000 (hexadecimal) for a 0 bit and write FFFFFFFF (hexadecimal) for a 1 bit.

SYNC MASK REGISTER (MASK)

Address: REG + 0x0418

Type: Write Only

Register Descriptions:

The Sync Mask Register (MASK) is a 16-bit serially programmable register. The mask allows the user to designate “no compare” bit positions in the 64-bit sync pattern. A zero in any bit position will result in a no compare for that location. If all of the bits in the sync pattern are to be used, the Sync Mask Register should be loaded with all ones. The sync pattern mask is programmed serially by performing 64 writes to the Sync Mask Register.

Bits are written out MSB first. Write 00000000 (hexadecimal) for a 0 bit and write FFFFFFFF (hexadecimal) for a 1 bit.

THRESHOLD REGISTER (THOLD)

Address: REG + 0x041C

Type: Write Only

Register Descriptions:

The Threshold Register (THOLD) is a 16-bit register. The table below defines the bit fields.

Table 4-15: Threshold Register (THOLD)

Bit	Name	Description
15:8	FTHOLD <7:0>	Flywheel/Lock Error Threshold for the input correlator is set by writing an 8-bit value <0 to 64> to the MSB of the THOLD Register. Whenever the total number of bits exceeds the threshold, the sync pattern is considered acceptable. For example, 31 = no bit errors allowed; 28 = 3 bit errors allowed in the sync pattern.
7:0	CTHOLD <7:0>	Search/Check Threshold for the input correlator is set by writing an 8-bit value <0 to 64> to the LSB of the THOLD Register. Whenever the number of bits exceeds the threshold, the sync pattern is considered acceptable. For example, 31 = no bit errors allowed; 28 = 3 bit errors allowed in the sync pattern.

RECEIVER CONTROL REGISTER 2 (RCR2)

Address: REG + 0x0420

Type: Write Only

Register Descriptions:

The Receiver Control Register 2 (RCR2) is a 16-bit register. The table below defines the bit fields.

Table 4-16: Receiver Control Register 2 (RCR2)

Bit	Name	Description
15	STAT_EN	A 1 enables the status structure to be appended to the end of each frame.
14	TMS_MFB	Set to 0 for one TMS strobe and interrupt for each frame. Set to 1 for one TMS strobe and interrupt for every 16 frames.
13	TMS_SEL	A 1 enables the External Reference Clock. A 0 uses a divided version of the PCI clock.
12	TMS_FIRST	Set to 1 for time tag rising edge of first bit. Set to 0 for time tag falling edge of last bit.
11	TMS_STROBE	A 1 enables the TMS strobe output.
10:4		Reserved.
3:0	BPWD <3:0>	Bits per Word. Assigns the number of bits minus one in each data word.

RECEIVER RANDOMIZER STARTING OFFSET REGISTER (RX_RDLEN)

Address: REG + 0x0424

Type: Write Only

Register Descriptions:

The Receiver Randomizer Starting Offset Register (RX_RDLEN) is a 16-bit register. This register defines the number of bytes at the beginning of each frame which is not randomized. This can be used to preserve a non-randomized sync marker in the incoming data. This register should be programmed to a value given by the following equation.

RDLEN = Frame Length – Number of Non-Randomized Bytes

RECEIVER CRC STARTING OFFSET REGISTER (RX_CDLEN)

Address: REG + 0x0428

Type: Write Only

Register Descriptions:

The Receiver CRC Starting Offset Register (RX_CDLEN) is a 16-bit register. This register defines the number of data words at the beginning of each frame which should not be included in the CRC verification calculation. This can be used to exclude the sync marker from the calculation. The register should be programmed to a value given by the following equation.

CDLEN = Frame Length – Number of Excluded Words

RECEIVER CRC ENDING OFFSET REGISTER (RX_DEND)

Address: REG + 0x042C

Type: Write Only

Register Descriptions:

The Receiver CRC Ending Offset Register (RX_DEND) is a 16-bit register. This register defines the position of the CRC parity word within the frame. This can be used to exclude Reed-Solomon parity data from the CRC verification calculation. This register should be programmed to a value given by the following equation.

DEND = Frame Length – CRC Offset

RECEIVER GLOBAL RESET REGISTER (RGRST)

Address: REG + 0x043C

Type: Write Only

Register Descriptions:

The Receiver Global Reset Register (RGRST) is a 16-bit register. Writing any value to this register will reset the serial input portion of the MONARCH-E board. All registers will be set to 0, and the logic will return to an idle state.

RECEIVER STATUS REGISTER 0 (RSTAT0)

Address: REG + 0x0400

Type: Read Only

Register Descriptions:

The Receiver Status Register 0 (RSTAT0) is a 16-bit register. The table below defines the bit fields.

Table 4-17: Receiver Status Register 0 (RSTAT0)

Bit	Name	Description
15	RIRQ	Receive Interrupt Request. This bit indicates that the serial input channel is asserting its IRQ* output. The source of the interrupt request is determined by the remaining status bits. A 1 indicates that the RIRQ is asserted.
14	RSEC_IRQ	Reed-Solomon Interrupt Request. This bit indicates that the Reed-Solomon Error Correction chip is generating an interrupt request. A 1 indicates that the RSEC_IRQ is asserted.
13	CLK_FLG	Clock Flag. This bit is set by the rising edge of the receive clock. It can be periodically checked and cleared to determine if an input clock is present. This bit is cleared by writing to the CLK_ACK bit in the Receiver Command register.
12	APC_FLG	Autopolarity Correct Flag. This bit is set by the serial input control logic locking to an inverted sync pattern.
11:9	SDIS <2:0>	Slip Distance. This field indicates the magnitude and direction of the previous frame's slip window. The distance in bit periods is given by SDIS <2:0> - SLIPSEL <2:0>. For example, if the allowable slip window (SLIPSEL) is 1 and SDIS =1, no slip occurred. If SLIPSEL=1 and SDIS=0, the sync pattern occurred one bit later than expected.
8	CRC_ERR_FLG	CRC Error Flag. This bit is set if the value calculated by the CRC Decoder does not match the CRC field in the data.
7		Reserved.
6	UNLOCK_FLG	Unlock Flag. This bit indicates that the serial input channel has entered the SEARCH state from the LOCK or FLYWHEEL state (dropped frame lock). This bit is cleared by writing to the UNLOCK_ACK bit in the Receiver Command Register.
5	WIN_END_FLG	Window End Flag. This bit is set at the end of each input frame. It is cleared by writing to the WIN_END_ACK bit in the Receiver Command Register.
4	SLIP_FLG	Slip Flag. This bit indicates that a bit slip has been detected. This bit is cleared by writing to the SLIP_ACK bit in the Receiver Command Register.
3		Reserved.

Bit	Name	Description
2	LOCK_FLG	Lock Flag. This bit indicates that the logic has entered the LOCK state from the SEARCH or CHECK state. It is cleared by writing to the LOCK_ACK bit in the Receiver Command Register.
1:0	SICQ <1:0>	This 2-bit field indicates the state of the serial input channel control logic as shown in the table below: SICQ<1:0> Input Channel State 00 SEARCH 01 CHECK 10 FLYWHEEL 11 LOCK

RECEIVER STATUS REGISTER 1 (RSTAT1)

Address: REG + 0x0404

Type: Read Only

Register Descriptions:

The Receiver Status Register 1 (RSTAT1) is a 16-bit register. The table below defines the bit fields.

Table 4-18: Receiver Status Register 1 (RSTAT1)

Bit	Name	Description
15:10		Reserved.
9	TMS_FLG	Time Stamp Flag.
8		Reserved.
7	PCIOVRN_FLG	PCI FIFO Overrun Flag. This bit indicates that the serial input has attempted to write data into the FIFO (which feeds the PCI bus) when it is full. The bit is cleared by writing to the PCIOVRN_ACK bit in the Receiver Command Register. Be aware that this condition indicates some data has been lost.
6	PCIFF*	PCI FIFO Buffer Full Flag. This bit is 0 (active low) when the PCI FIFO buffer is full.
5	PCIHF*	PCI FIFO Buffer Half Full Flag. This bit is 0 (active low) when the PCI FIFO buffer is half full or greater.
4	PCIEF*	PCI FIFO Buffer Empty Flag. This bit is 0 (active low) when the PCI FIFO buffer is empty.
3	RSDOVRN_FLG	Reed-Solomon Decoder Overrun Flag. This bit indicates that the serial input has attempted to write data into the FIFO (which feeds the Reed-Solomon Decoder) when it is full. This bit is cleared by writing to the RSDOVRN_ACK bit in the Receiver Command Register. Be aware that this condition indicates some data has been lost.
2	RSDFF*	Reed-Solomon Decoder FIFO Buffer Full Flag. This bit is 0 (active low) when the Reed-Solomon Decoder FIFO buffer is full.
1	RSDHF*	Reed-Solomon Decoder FIFO Buffer Half Full Flag. This bit is 0 (active low) when the Reed-Solomon Decoder FIFO buffer is half full or greater.
0	RSDEF*	Reed-Solomon Decoder FIFO Buffer Empty Flag. This bit is 0 (active low) when the Reed-Solomon Decoder FIFO buffer is empty.

Section 4.8
TRANSMITTER REGISTERS

The tables below list the different programmable registers for the Transmitter. All accesses to the Xilinx registers must be 32 bits wide, although not all of the bits may be significant. To ensure compatibility with future enhancements, all reserved bits should be set to 0 when writing to a register and ignored when reading from a register.

Table 4-19: Transmitter Register Map – Write Only

Offset	Register	Name
0	TCOM	Transmitter Command Register
4	TICR	Transmitter Interrupt Control Register
8	TCR	Transmitter Control Register
C	TOCR	Transmitter Output Control Register
10	TX_TFL	Transmitter Frame Length
14	TX_RDLEN	Transmitter Randomizer Starting Offset
18	TX_CDLEN	Transmitter CRC Starting Offset
1C	TX_DEND	Transmitter CRC Ending Offset
20	DDSCLK	Direct Digital Synthesizer Clock Register
24	STPLL	Transmit Clock Register
28	REFPLL	Reference Clock Register
2C	XCENC	RSENC FPGA Configuration Data
30	ERRCR	Error Control Register
34	ERRDAT	Error Data Register
38	TOCR2	Transmitter Output Control Register 2
3C	TGRST	Transmitter Global Reset

Table 4-20: Transmitter Register Map – Read Only

Offset	Register	Name
0	TSTAT0	Transmitter Status Register

TRANSMITTER COMMAND REGISTER (TCOM)

Address: REG + 0x0C00

Type: Write Only

Register Descriptions:

The Transmitter Command Register (TCOM) is a 16-bit register. The table below defines the bit fields.

Table 4-21: Transmitter Command Register (TCOM)

Bit	Name	Description
15		Reserved.
14	DDS_EN	Writing a 1 to this register transfers the frequency control word from the frequency register to the phase accumulator's input register.
13	CLK_ACK	Clock Acknowledge. Writing a 1 to this bit clears the CLK_FLG bit in the Transmitter Status Register.
12:11		Reserved.
10	INS_SLIP	Insert Slip will enter a bit slip in the data stream. The number of bits slipped is based on the value of SLIPSEL. The direction of the slip will toggle between forward and backward.
9	INS_ERR	Insert Error will insert a burst error starting at the offset specified by ERRDAT and a length as specified by EBLEN.
8:7		Reserved.
6	UNRN_ACK	Underrun Acknowledge. Writing a 1 to this bit clears the UNRN_FLG bit in the Transmitter Status Register.
5	EOF_ACK	End of Frame Acknowledge. Writing a 1 to this bit clears the EOF_FLG bit in the Transmitter Status Register.
4	FAF_ACK	FIFO Almost Full Acknowledge. Writing a 1 to this bit clears the FAF_FLG bit in the Transmitter Status Register.
3	LOD_ACK	Loss of Data Acknowledge. Writing a 1 to this bit clears the LOD_FLG bit in the Transmitter Status Register.
2		Reserved.
1	STOP	Writing a 1 to this bit issues a STOP command to the serial output control logic. This will cause the output control logic to return to the IDLE state.
0	START	Writing a 1 to this bit issues a START command to the serial output control logic. This will cause the output control logic to enter the READY state. Once data becomes available, it will be output.

TRANSMITTER INTERRUPT CONTROL REGISTER (TICR)

Address: REG + 0x0C04

Type: Write Only

Register Descriptions:

The Transmitter Interrupt Control Register (TICR) is a 16-bit register. The table below defines the bit fields.

Table 4-22: Transmitter Interrupt Control Register (TICR)

Bit	Name	Description
15:7		Reserved.
6	UNRN_IEN	Underrun Interrupt Enable. Writing a 1 allows the host processor to be interrupted whenever an underrun occurs.
5	EOF_IEN	End of Frame Interrupt Enable. Writing a 1 allows the host processor to be interrupted at the end of each frame.
4	FAF_IEN	FIFO Almost Full Interrupt Enable. Writing a 1 allows the host processor to be interrupted whenever the FIFO Almost Full is asserted.
3	LOD_IEN	Loss of Data Interrupt Enable. Writing a 1 allows the host processor to be interrupted whenever Loss of Data is detected.
2:0		Reserved.

Transmitter Control Register (TCR)

Address: REG + 0x0C08

Type: Write Only

Register Descriptions:

The Transmitter Control Register (TCR) is a 16-bit register. The table below defines the bit fields.

Table 4-23: Transmitter Control Register (TCR)

Bit	Name	Description
15	XCENC_PROG	Reed-Solomon Encoder FPGA Program. Initiates a configuration cycle for the FPGA.
14	REF_POT	Reference potentiometer. Writing a 0 selects the internal DDS reference. Writing a 1 selects an external DDS reference.
13	REF_INV	Inverts Reference Signal for external references only.
12:11	REFSEL <1:0>	This 2-bit field selects the Digital Direct Synthesizer's reference inputs shown in the table below: REFSEL<1:0> DDS Inputs 00 Reserved 01 Internal PLL 10 External TTL 11 External RS-422 *NOTE:* REF_POT must be set to 1 for REFSEL to govern.
10	X	Undefined.
9:8	STSEL <1:0>	This 2-bit field selects the send timing signals shown in the table below: STSEL<1:0> Send Timing Inputs 00 Internal DDS 01 Internal PLL 10 External TLL 11 External RS-422
7:2		Reserved.
1	REF_PLL_EN	Reference PLL Enable. A 1 selects the programmed frequency for the PLL. A 0 selects the default frequency.
0	ST_PLL_EN	Transmit Clock PLL Enable. A 1 selects the programmed frequency. A 0 selects the default frequency.

TRANSMITTER OUTPUT CONTROL REGISTER (TOCR)

Address: REG + 0x0C0C

Type: Write Only

Register Descriptions:

The Transmitter Output Control Register (TOCR) is a 16-bit register. The table below defines the bit fields.

Table 4-24: Transmitter Output Control Register (TOCR)

Bit	Name	Description
15	SYM_REV	Symbol Reverse. A 1 will reverse the symbols in the convolutional encoder.
14	SYMB_INV	Symbol Invert. A 1 will invert the second symbol in the convolutional encoder.
13	SYMA_INV	Symbol Invert. A 1 will invert the first symbol in the convolutional encoder.
12		Reserved.
11	CONV_EN	Convolutional Encoder Enable. A 1 will enable the convolutional encoder.
10:8	OCSEL <2:0>	Output Code Select. This 3-bit field identifies which of the different output codes, as shown in the table below, is used. This differential coding is performed after convolutional encoding (if enabled). OCSEL <2:0> — Output Codes 000 NRZ-L 001 NRZ-S 010 NRZ-M 011 Reserved 100 Biphase-L 101 Biphase-S 110 Biphase-M 111 Reserved
7	TD_INV	Transmit Data Invert. For RS-422, writing a 1 inverts the transmitter data. For TTL, writing a 0 inverts the transmitter data.
6	TC_INV	Transmit Clock Invert. For RS-422, writing a 1 inverts the transmitter clock. For TTL, writing a 0 inverts the transmitter clock.
5	GATE_MODE	Writing a 1 preserves the output clock when no valid data exists. Writing a 0 will clamp the clock low when no data exists.
4	RAND_EN	Randomizer Enable. Writing a 1 enables the Randomizer.
3	CRC_EN	CRC Enable. Writing a 1 enables CRC verification.
2	RAND_MODE	This bit selects whether the Randomizer offset is used. Writing a 1 disables the offset. A 0 selects the offset.
1	CRC_MODE	This bit selects whether the CRC offset is used. Writing a 1 disables the offset. A 0 selects the offset.

Bit	Name	Description
0	RAW_MODE	Set to 0 for fixed size frames. Set to 1 to transmit variable size blocks of data (Reed-Solomon encoder, CRC encoder, and Randomizer must be disabled in raw mode).

TRANSMITTER FRAME LENGTH REGISTER (TX_TFL)

Address: REG + 0x0C10

Type: Write Only

Register Descriptions:

The Transmitter Frame Length Register (TX_TFL) is a 16-bit register that indicates the number of words in each frame. It should be programmed to the desired frame length minus one. The 16-bit value is loaded into the octet counter at the start of each frame.

TRANSMITTER RANDOMIZER STARTING OFFSET REGISTER (TX_RDLEN)

Address: REG + 0x0C14

Type: Write Only

Register Descriptions:

The Transmitter Randomizer Starting Offset Register (TX_RDLEN) is a 16-bit register that defines the number of bytes at the beginning of each frame which is not randomized. This can be used to preserve a non-randomized sync marker. This register should be programmed to a value given by the following equation.

RDLEN = Frame Length – Number of Non-Randomized Bytes

TRANSMITTER CRC STARTING OFFSET REGISTER (TX_CDLEN)

Address: REG + 0x0C18

Type: Write Only

Register Descriptions:

The Transmitter CRC Starting Offset Register (TX_CDLEN) is a 16-bit register that defines the number of data words at the beginning of each frame which should not be included in the CRC verification calculation. This can be used to exclude the sync marker from the calculation. The register should be programmed to a value given by the following equation.

CDLEN = Frame Length – Number of Excluded Words

TRANSMITTER CRC ENDING OFFSET REGISTER (TX_DEND)

Address: REG + 0x0C1C

Type: Write Only

Register Descriptions:

The Transmitter CRC Ending Offset Register (TX_DEND) is a 16-bit register that defines the position of the CRC parity word within the frame. This can be used to exclude Reed-Solomon parity data from the CRC encoding calculation. If CRC insertion is enabled, the calculated CRC parity will be inserted into the outgoing data starting at this offset. The register should be programmed to a value given by the following equation.

DEND = Frame Length – CRC Offset

DIRECT DIGITAL SYNTHESIZER CLOCK REGISTER (DDSCLK)

Address: REG + 0x0C20

Type: Write Only

Register Descriptions:

The Direct Digital Synthesizer Clock Register (DDSCLK) is a serial register used to program the DDS clock frequency. The 32-bit DDS codeword must be written to this register MSB first. Write 00000000 (hexadecimal) for a 0 bit and write FFFFFFFF (hexadecimal) for a 1 bit. Write a 1 to TCOM's DDS_EN (bit 14) to transfer the frequency control word to the phase accumulator's input register.

See the DDS Clock Generator (HSP45102) data sheet for a detailed description of the codeword calculation.

PLL TRANSMIT CLOCK REGISTER (STPLL)

Address: REG + 0x0C24

Type: Write Only

Register Descriptions:

The PLL Transmit Clock Register (STPLL) is a serial register used to program the transmit clock portion of the PLL clock generator chip. The 22-bit codeword must be written to this register LSB first.

See the PLL Clock Generator (ICD2051) data sheet for a detailed description of the codeword calculation.

PLL Reference Clock Register (REFPLL)

Address: REG + 0x0C28

Type: Write Only

Register Descriptions:

The PLL Reference Clock Register (REFPLL) is a serial register used to program the reference portion of the PLL clock generator chip. The 22-bit codeword must be written to this register LSB first.

See the PLL Clock Generator (ICD2051) data sheet for a detailed description of the codeword calculation.

RSENC FPGA Configuration Data Register (XCENC)

Address: REG + 0x0C2C

Type: Write Only

Register Descriptions:

The RSENC FPGA Configuration Data Register (XCENC) is a serial register used to send configuration data to the RSENC Xilinx FPGA. Once configuration mode has been entered, each byte of the configuration file must be written to this register one bit at a time starting with the LSB.

TRANSMITTER ERROR CONTROL REGISTER (ERRCR)

Address: REG + 0x0C30

Type: Write Only

Register Descriptions:

The Transmitter Error Control Register (ERRCR) is a 16-bit register. The table below defines the bit fields.

Table 4-25: Transmitter Error Control Register (ERRCR)

Bit	Name	Description
15:10		Reserved.
9:8	SLIPSEL <1:0>	Defines the number of bits to slip when INS_SLIP bit in the Transmitter Command Register is asserted.
7:0	EBLEN <7:0>	Error Bit Burst Length defines the number of bits in burst error to insert when INS_ERR bit in the Transmitter Command Register is asserted.

TRANSMITTER ERROR DATA REGISTER (ERRDAT)

Address: REG + 0x0C34

Type: Write Only

Register Descriptions:

The Transmitter Error Data Register (ERRDAT) is a 16-bit register that is used to specify the offset within the frame of a data burst error. When the INS_ERR bit in the Transmitter Command Register is set, data will continue to be output as normal until the specified offset is reached. Data will then be inverted for the number of bits specified by EBLEN in the Error Control Register. Once the specified burst has been inserted, data will return to normal. This register should be set as specified by the following equation.

ERRDAT = Frame Length – Error Offset

TRANSMITTER OUTPUT CONTROL REGISTER 2 (TOCR2)

Address: REG + 0x0C38

Type: Write Only

Register Descriptions:

The Transmitter Output Control Register 2 (TOCR2) is a 16-bit register. The table below defines the bit fields. This differential coding is performed prior to convolutional encoding (if enabled).

Table 4-26: Transmitter Output Control Register 2 (TOCR2)

Bit	Name	Description
15:2		Reserved.
1:0	POCSEL <1:0>	Pre-output Code Select, where 0 = NRZ-L, 1 = NRZ-S, 2 = NRZ-M, and 3 is reserved.

TRANSMITTER GLOBAL RESET REGISTER (TGRST)

Address: REG + 0x0C3C

Type: Write Only

Register Descriptions:

Writing any value to the Transmitter Global Reset Register (TGRST) will reset the serial output portion of the MONARCH-E board. All registers will be set to 0, and the logic will return to an idle state.

TRANSMITTER STATUS REGISTER (TSTAT0)

Address: REG + 0x0C00

Type: Read Only

Register Descriptions:

The Transmitter Status Register (TSTAT0) is a 16-bit register. The table below defines the bit fields.

Table 4-27: Transmitter Status Register (TSTAT0)

Bit	Name	Description
15	TIRQ	Transmitter Interrupt Request. This bit indicates the serial output channel is asserting its IRQ* output. The source of the interrupt request is determined by the remaining status bits. A 1 indicates that TIRQ is asserted.
14		Reserved.
13	CLK_FLG	Clock Flag. This bit is set by the rising edge of the transmit clock. It can be periodically checked and cleared to determine if an output clock is present. This bit is cleared by writing to the CLK_ACK bit in the Transmitter Command Register.
12	XCENC_INIT*	Reed-Solomon Encoder FPGA Initialization. A 1 indicates that the Reed-Solomon FPGA is ready to accept configuration data.
11	XCENC_DONE	Reed-Solomon Encoder FPGA Configuration Complete. A 1 indicates that the REED-SOLOMON FPGA has been successfully configured.
10	SLIPDIR	Slip Direction. This bit indicates the direction of the slip which will be inserted the next time the INS_SLIP bit in the Transmitter Command Register is asserted. A 0 indicates a backward slip while a 1 indicates forward slip.
9:8	SOCQ <1:0>	This 2-bit field indicates the state of the serial output channel control logic as shown in the table below: SOCQ <1:0> — Output Channel State 00 — IDLE 01 — READY 10 — ERROR 11 — RUN
7		Reserved.
6	UNRN_FLG	Underrun Flag.
5	EOF_FLG	End of Frame Flag. This bit is set at the end of each frame.
4	FAF_FLG	FIFO Almost Full Flag. This bit indicates that the FIFO is Almost Full.
3	LOD_FLG	Loss of Data Flag. This bit indicates that Loss of Data has occurred. (FIFO becomes empty on a frame boundary)

Bit	Name	Description
2	SOAF*	Serial Output FIFO Buffer Almost Full Flag. This bit is 0 (active low) when the output FIFO buffer is almost full.
1	SOHF*	Serial Output FIFO Buffer Half Full Flag. This bit is 0 (active low) when the output FIFO buffer is half full or greater.
0	SOEF*	Serial Output FIFO Buffer Empty Flag. This bit is 0 (active low) when the output FIFO buffer is empty.

Section 4.9
REED-SOLOMON ENCODER REGISTERS

The table below lists the different programmable registers for the Reed-Solomon Encoder.

Table 4-28: Reed-Solomon Encoder Register Map – Write Only

Offset	Register	Name
0	RSCR1	Reed-Solomon Encoder Control Register 1
4	RSCR2	Reed-Solomon Encoder Control Register 2
8	PREAM	Preamble Length
C	RS_TFL	Reed Solomon Frame Length

REED-SOLOMON ENCODER CONTROL REGISTER 1 (RSCR1)

Address: REG + 0x0800

Type: Write Only

Register Descriptions:

The Reed-Solomon Encoder Control Register 1 (RSCR1) is a 16-bit register. The table below defines the bit fields.

Table 4-29: Reed-Solomon Encoder Control Register 1 (RSCR1)

Bit	Name	Description
15:10		Reserved.
9	RSRAW_MODE	Set to 0 for fixed size frames. Set to 1 to transmit variable size blocks of data (Reed-Solomon encoder, CRC encoder, and Randomizer must be disabled in raw mode).
8	BURST_EN	Burst Enable. If set to 1, the FIFO is required to be at least half full when transmission begins. If set to 0, any data in the FIFO will start transmission.
7	IDLE_DAT	Idle Data. This bit governs the transmitter's behavior if no data is present. If set to a 1, the data line will be held at 0. If set to a 0, data will alternate between 1 and 0.
6	LSB_MSB	This bit controls the order in which parallel data is converted to serial data. Writing a 1 to this bit indicates LSB first. A 0 indicates MSB first.
5	BSWAP_EN	Byte Swap. Writing a 1 to this bit will perform a 32-bit byte swap of the incoming parallel data. **This bit should be set to 1 for x86 (Little Endian) platforms.**
4		Reserved.
3:0	BPWD <3:0>	Bits Per Word. This field defines the number of bits per word minus one.

REED-SOLOMON ENCODER CONTROL REGISTER 2 (RSCR2)

Address: REG + 0x0804

Type: Write Only

Register Descriptions:

The Reed-Solomon Encoder Control Register 2 (RSCR2) is a 16-bit register. The table below defines the bit fields.

Table 4-30: Reed-Solomon Encoder Control Register 2 (RSCR2)

Bit	Name	Description
15:11		Reserved.
10	RS_OVWR	Reed-Solomon Encoder Parity Overwrite. Writing a 1 to this field indicates that the Reed-Solomon parity data will overwrite the frame data. If this field contains a 0, the Reed-Solomon parity data will be appended to the end of each frame.
9	PREAM_EN	Preamble Enable. Writing a 1 to this field indicates that the Reed-Solomon Encoding should ignore the preamble. If this field contains a 0, the frame contains no preamble.
8	RS_EN	Reed-Solomon Encoder Enable. Writing a 1 to this field indicates that parity data will be calculated and added to each frame.
7:3		Reserved.
2:0	I <2:0>	Interleave level. This field defines the interleave level minus one used by the Reed-Solomon Encoder.

PREAMBLE LENGTH REGISTER (PREAM)

Address: REG + 0x0808

Type: Write Only

Register Descriptions:

The Preamble Length Register (PREAM) is a 16-bit register that defines the number of data words at the beginning of each frame to exclude from the Reed-Solomon parity calculation. This can be used to skip the sync marker. This register should be programmed to a value given by the following equation.

PREAM = Frame Length – Number of Excluded Words

REED-SOLOMON TRANSMITTER FRAME LENGTH REGISTER (RS_TFL)

Address: REG + 0x080C

Type: Write Only

Register Descriptions:

The Reed-Solomon Transmitter Frame Length Register (RS_TFL) is a 16-bit register that indicates the number of words in each frame. It should be programmed to the desired frame length minus one. The 16-bit value is loaded into the octet counter at the start of each frame.

The value should be the same as the value specified in the **Transmitter Frame Length Register (TX_TFL)** (REG + 0x0C10), except in raw mode, this value should be set to 0xFFFF.

Chapter 5
Signal Descriptions

This section contains the DSUB 37-pin connector and PCI bus signal descriptions for the MONARCH-E. The DSUB 37-pin connector P3 provides RS-422 and TTL level clock and data serial I/O signals. The PCI gold fingers provide standard I/O signals to communicate between the PCI bus and add-on card.

Section 5.1
CONFIGURATION JUMPERS

Jumpers J6 and J7 are connected to PCI signals PRSNT1 and PRSNT2. They are used to indicate the maximum power requirements of the board. For the MONARCH-E, J6 should have a shunt jumper and J7 should be left open. Jumpers J1 through J5 are not used and should be left open.

Section 5.2
STATUS LEDS

The three LEDs on the board, LED1, LED2, and LED3, are used to indicate the state of the Xilinx FPGAs, U10, U11, and U14 respectively. The LED will be off if its corresponding FPGA has not been configured and lit if it has been.

Section 5.3
PCI BUS GOLD FINGERS

The PCI bus gold fingers interconnect with a 32-bit, 5V PCI slot on the host motherboard. The pin assignments, signal mnemonics, and signal descriptions for the PCI bus gold fingers are shown in **Table 5-1** and **Table 5-2** respectively.

Table 5-1: PCI Bus Gold Fingers

5 Volt Board								
Pin	**Side B**	**Side A**	**Pin**	**Side B**	**Side A**	**Pin**	**Side B**	**Side A**
1	-12V	TRST#	**21**	AD[29]	+3.3V	**42**	SERR#	Ground
2	TCK	+12V	**22**	Ground	AD[28]	**43**	+3.3V	PAR
3	Ground	TMS	**23**	AD[27]	AD[26]	**44**	C/BE[1]#	AD[15]
4	TDO	TDI	**24**	AD[25]	Ground	**45**	AD[14]	+3.3V
5	+5V	+5V	**25**	+3.3V	AD[24]	**46**	Ground	AD[13]
6	+5V	INTA#	**26**	C/BE[3]#	IDSEL	**47**	AD[12]	AD[11]
7	INTB#	INTC#	**27**	AD[23]	+3.3V	**48**	AD[10]	Ground
8	INTD#	+5V	**28**	Ground	AD[22]	**49**	Ground	AD[09]
9	PRSNT1#	Reserved	**29**	AD[21]	AD[20]	**50** **51**	KEYWAY KEYWAY	
10	Reserved	+5V	**30**	AD[19]	Ground	**52**	AD[08]	C/BE[0]#
11	PRSNT2#	Reserved	**31**	+3.3V	AD[18]	**53**	AD[07]	+3.3V
12	Ground	Ground	**32**	AD[17]	AD[16]	**54**	+3.3V	AD[06]
13	Ground	Ground	**33**	C/BE[2]#	+3.3V	**55**	AD[05]	AD[04]
14	Reserved	Reserved	**34**	Ground	FRAME#	**56**	AD[03]	Ground
15	Ground	RST#	**35**	IRDY#	Ground	**57**	Ground	AD[02]
16	CLK	+5V	**36**	+3.3V	TRDY#	**58**	AD[01]	AD[00]
17	Ground	GNT#	**37**	DEVSEL#	Ground	**59**	+5V	+5V
18	REQ#	Ground	**38**	Ground	STOP#	**60**	ACK64#	REQ64#
19	+5V	Reserved	**39**	LOCK#	+3.3V	**61**	+5V	+5V
20	AD[31]	AD[30]	**40**	PERR#	SDONE	**62**	+5V	+5V
			41	+3.3V	SBO#			

The following conventions are used **Table 5-2**, PCI Signal Description:

- in *Input* is a standard input-only signal.
- out *Totem Pole Output* is a standard active driver.
- t/s *Tri-State®* is a bi-directional, tri-state input/output pin.

- s/t/s *Sustained Tri-State* is an active low tri-state owned and driven by one and only one agent at a time. The agent that drives an s/t/s pin low must drive it high for at least one clock before letting it float. A new agent cannot start driving an s/t/s signal any sooner than one clock after the previous owner tri-states it. A pull-up is required to sustain the inactive state until another agent drives it, and must be provided by the central resource.

- o/d *Open Drain* allows multiple devices to share as a wire-OR.

Table 5-2: PCI Bus Signal Descriptions

Signal Name	Signal Type	Signal Description
CLK	in	*Clock* provides timing for all transactions on PCI and is an input to every PCI device. All other PCI signals, except RST#, IRQA#, IRQB#, IRQC#, and IRQD#, are sampled on the rising edge of CLK, and all other timing parameters are defined with respect to this edge. PCI operates up to 33 MHz and, in general, the minimum frequency is DC (0 Hz).
RST#	in	*Reset* is used to bring PCI-specific registers, sequencers, and signals to a consistent state. Anytime RST# is asserted, all PCI output signals must be driven to their benign states. In general, this means they must be tri-stated. SERR# (open drain) is floated. SBO# and SDONE may optionally be driven to a logic low level if tri-state outputs are not provided here. REQ# and GNT# must both be tri-stated (they cannot be driven low or high during reset). To prevent AD, C/BE#, and PAR signals from floating during reset, the central device may drive these lines during reset (bus parking) but only to a logic low level; they may not be driven high. RST# may be asynchronous to CLK when asserted or de-asserted. Although asynchronous, de-assertion is guaranteed to be a clean, bounce-free edge. Except for configuration accesses, only devices that are required to boot the system will respond after reset.
ADDRESS AND DATA PINS		
AD[31::00]	t/s	*Address* and *Data* are multiplexed on the same PCI pins. A bus transaction consists of an address phase followed by one or more data phases. PCI supports both read and write bursts. The address phase is the clock cycle in which FRAME# is asserted. During the address phase AD[31::00] contain a physical address (32 bits). For I/O, this is a byte address; for configuration and memory, it is a DWORD address. During data phases, AD[07::00] contain the least significant byte (LSB) and AD[31::24] contain the most significant byte (MSB). Write data is stable and valid when IRDY# is asserted, and read data is stable and valid when TRDY# is asserted. Data is transferred during those clocks where both IRDY# and TRDY# are asserted.
C/BE[3::0]#	t/s	*Bus Command* and *Byte Enables* are multiplexed on the same PCI pins. During the address phase of a transaction, C/BE[3::0]# define the bus command. During the data phase, C/BE[3::0]# are used as Byte Enables. The Byte Enables are valid for the entire data phase and determine which byte lanes carry meaningful data. C/BE[0]# applies to byte 0 (LSB) and C/BE[3]# applies to byte 3 (MSB).

Signal Name	Signal Type	Signal Description
PAR	t/s	*Parity* is even parity across AD[31::00] and C/BE[3::0]#. Parity generation is required by all PCI agents. PAR is stable and valid one clock after the address phase. For data phases, PAR is stable and valid one clock after either IRDY# is asserted on a write transaction or TRDY# is asserted on a read transaction. Once PAR is valid, it remains valid until one clock after the completion of the current data phase. (PAR has the same timing as AD[31::00] but is delayed by one clock.) The master drives PAR for address and write data phases; the target drives PAR for read data phases.
INTERFACE CONTROLS PINS		
FRAME#	s/t/s	*Cycle Frame* is driven by the current master to indicate the beginning and duration of an access. FRAME# is asserted to indicate a bus transaction is beginning. While FRAME# is asserted, data transfers continue. When FRAME# is de-asserted, the transaction is in the final data phase.
IRDY#	s/t/s	*Initiator Ready* indicates the initiating agent's (bus master's) ability to complete the current data phase of the transaction. IRDY# is used in conjunction with TRDY#. A data phase is completed on any clock where both IRDY# and TRDY# are sampled asserted. During a write, IRDY# indicates that valid data is present on AD[31::00]. During a read, it indicates the master is prepared to accept data. Wait cycles are inserted until both IRDY# and TRDY# are asserted together.
TRDY#	s/t/s	*Target Ready* indicates the target agent's (selected device's) ability to complete the current data phase of the transaction. TRDY# is used in conjunction with IRDY#. A data phase is completed on any clock where both TRDY# and IRDY# are sampled asserted. During a read, TRDY# indicates that valid data is present on AD[31::00]. During a write, it indicates the target is prepared to accept data. Wait cycles are inserted until both IRDY# and TRDY# are asserted together.
STOP#	s/t/s	*Stop* indicates the current target is requesting the master to stop the current transaction.
LOCK#	s/t/s	*Lock* indicates an atomic operation that may require multiple transactions to complete. When LOCK# is asserted, non exclusive transactions may proceed to an address that is not currently locked. A grant to start a transaction on PCI does not guarantee control of LOCK#. Control of LOCK# is obtained under its own protocol in conjunction with GNT#. It is possible for different agents to use PCI while a single master retains ownership of LOCK#.
IDSEL	in	*Initialization Device* Select is used as a chip select during configuration read and write transactions.
DEVSEL#		*Device Select*, when actively driven, indicates the driving device has decoded its address as the target of the current access. As an input, DEVSEL# indicates whether any device on the bus has been selected.
ARBITRATION PINS (BUS MASTERS ONLY)		
REQ#	t/s	*Request,* indicates to the arbiter that this agent desires use of the bus. This is a point-to-point signal. Every master has its own REQ#.
GNT#	t/s	*Grant* indicates to the agent that access to the bus has been granted. This a point-to-point signal. Every master has its own GNT#.

Signal Name	Signal Type	Signal Description
PERR#	s/t/s	*Parity Error* is used only for the reporting of data parity errors during all PCI transactions except a Special Cycle. The PERR# pin is sustained tri-state and must be driven active by the agent receiving data two clocks following the data when a data parity error is detected. The minimum duration of PERR# is one clock for each data phase that a data parity error is detected. (If sequential data phases each have a data parity error, the PERR# signal will be asserted for more than a single clock.) PERR# must be driven high for one clock before being tri-stated as with all sustained tri-state signals. There are no special conditions when a data parity error may be lost or when reporting an error may be delayed. An agent cannot report a PERR# until it has claimed the access by asserting DEVSEL# and completed a data phase.
SERR#	o/d	*System Error* is for reporting address parity errors, data parity errors on the Special Cycle command, or any other system error where the result will be catastrophic. If an agent does not want a non-maskable interrupt (NMI) to be generated, a different reporting mechanism is required. SERR# is pure open drain and is actively driven for a single PCI clock by the agent reporting the error. The assertion of SERR# is synchronous to the clock and meets the setup and hold times of all bused signals. However, the restoring of SERR# to the de-asserted state is accomplished by a weak pull-up (same value as used for s/t/s) which is provided by the system designer and not by the signaling agent or central resource. This pull-up may take two to three clock periods to fully restore SERR#. The agent that reports SERR#s to the operating system does so anytime SERR# is sampled asserted.

Appendix A:

CONFIGURATION AND INSTALLATION DRAWINGS

Note: The following pages are best viewed when printed on 11” x 17” (size B) paper, set up for landscape orientation.

NOTES:

1. THE FOLLOWING DOCUMENT SUPPLEMENTS THE CONFIGURATION FILE. SEE THE CONFIGURATION FILE FOR SPECIFIC SETTING BASED ON SLOT LOCATION.

2. THE FOLLOWING DOCUMENT PROVIDES DETAILED INFORMATION ON HOW TO CONFIGURE THE AVTEC MONARCH (E) CCA. THIS DOCUMENT IN ADDITION WILL BE A SINGLE SOURCE OF REFERENCE FOR SWITCH SETTINGS, JUMPER CONFIGURATIONS, AND PROVIDE THE INSTALLER WITH CONNECTOR INTERFACE PINOUTS.

3. CAUTION – THIS ITEM CONTAINS PARTS SENSITIVE TO DAMAGE BY ELECTROSTATIC DISCHARGE (ESD). USE PRECAUTIONARY PROCEDURES WHEN TOUCHING, REMOVING, OR INSERTING PARTS OR ASSEMBLIES.

4. REFERENCE DESIGNATIONS ARE USED IN PLACE OF FIND NUMBERS FOR LOCATING COMPONENTS WHERE APPLICABLE.

5. SQUARE PAD DENOTES PIN 1.

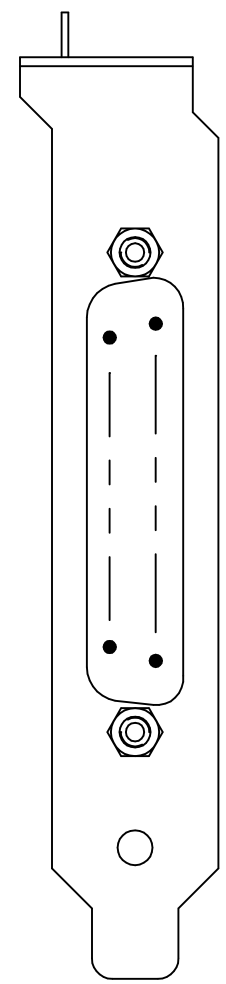
6. CONFIGURE JUMPERS AS SHOWN ON SHEET 2.

7. LED INDICATIONS AS NOTED ON SHEET 2.

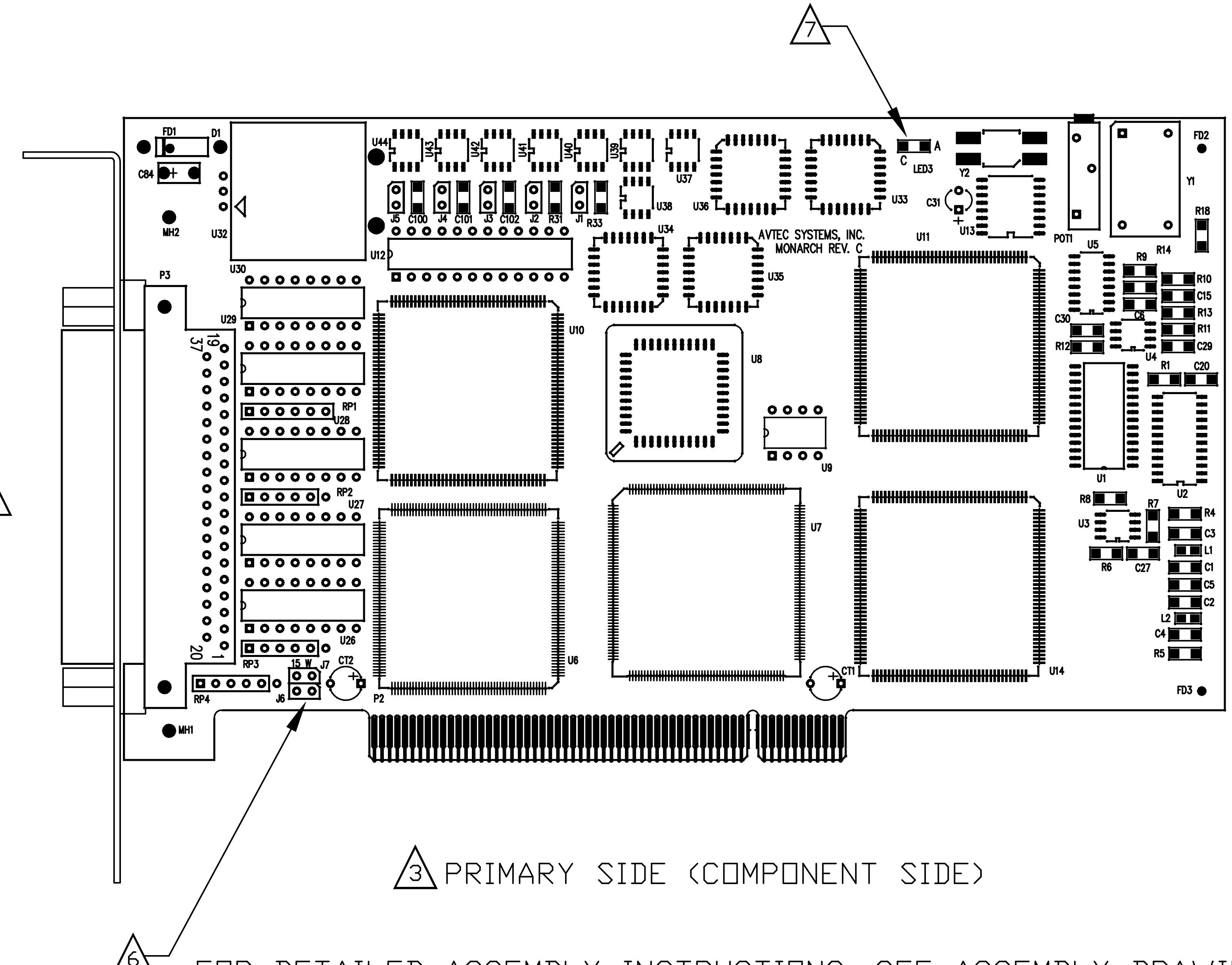
8. THE P3 CONNECTOR I/O PINOUTS ARE SHOWN ON SHEET 3. CONNECTOR IS DB37 – RECEPTACLE (FEMALE).

9. AVTEC'S PRODUCTS AND TECHNOLOGY ARE CONTROLLED FOR EXPORT PURPOSES BY THE U.S. GOVERNMENT PURSUANT TO THE ARMS EXPORT CONTROL ACT AND THE INTERNATIONAL TRAFFIC IN ARMS REGULATIONS OR THE EXPORT ADMINISTRATION ACT AND THE EXPORT ADMINISTRATION REGULATIONS.

REVISIONS				
ZONE	LTR	DESCRIPTION	DATE	APPD



8



3 PRIMARY SIDE (COMPONENT SIDE)

FOR DETAILED ASSEMBLY INSTRUCTIONS, SEE ASSEMBLY DRAWING

ITEM NO.	QTY REQD -4	QTY REQD -3	QTY REQD -2	QTY REQD -1	CODE IDENT	PART OR IDENTIFYING NO.	SPECIFICATION	NOMENCLATURE OR DESCRIPTION	REF DESIG	NOTE NO.

NEXT ASSY	USED ON
APPLICATION	

FINISH: N/A

UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES
TOLERANCES:
2 PLACES DEC. ±.01
3 PLACES DEC. ±.005
ANGLES ± °
MATERIAL: N/A

CONTR NO.		
DR	Myron P. Szot	1/21/1999
HDWR ENG.	Mike Williams	1/23/1999
SYS ENG.	Mike Williams	1/23/1999
OPS MGR	Walt Galvin	1/23/1999
PROJ APPD	Roy Delawder	1/23/1999

AVTEC SYSTEMS
10530 ROSEHAVEN STREET, SUITE 300
FAIRFAX, VA 22030

TITLE: MONARCH (E) CONFIG. AND INSTALL

SIZE	CAGE	DWG NO.	REV
B	6X005	PCI-6X00-CI	C

SCALE: NONE | SHEET 1 OF 3

TYPE DOCUMENT	IDENT NO.	DESCRIPTION
REFERENCED DOCUMENTS		

REVISIONS				
ZONE	LTR	DESCRIPTION	DATE	APPD

FIGURE 1: JUMPER SETTINGS

FOR ALL VERSIONS OF THE MONARCH CCA, CONFIGURE JUMPERS AS SHOWN BELOW (ALL OPTIONS DETAILED IN TABLE 1):

1) J6 IS INSTALLED PER THE PCI SPECIFICATION (SIGNAL 'PRSNT1#' IS USED TO INDICATE THE MAXIMUM POWER REQUIREMENTS OF THE CCA. IN THIS CASE, IT IS SET FOR 25 WATTS MAXIMUM.)
2) J7 IS OPEN (NO JUMPERS INSTALLED ON THE HEADER)
3) J1 THROUGH J5 ARE NOT USED THEREFORE THEY ARE OPEN (NO JUMPERS INSTALLED; DESIGNED FOR EXPANSION)

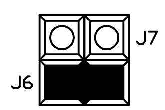


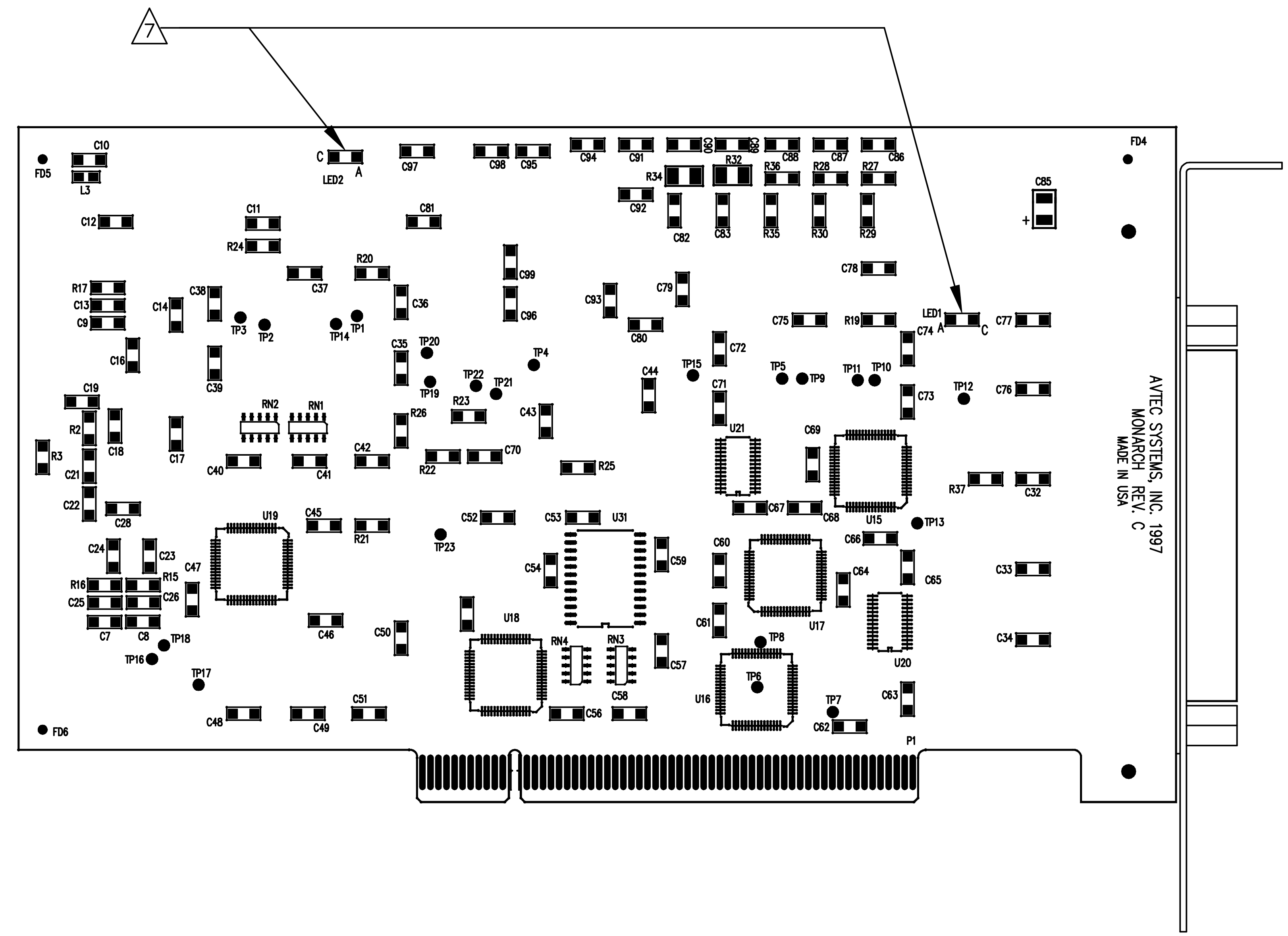
TABLE 1: OPTIONAL JUMPER SETTINGS

J6 & J7 JUMPER PIN ASSIGNMENTS (IAW PCI SPEC)		
J6 JUMPER (PRSNT1#)	J7 JUMPER (PRSNT2#)	SIGNAL DESCRIPTION
NONE	NONE	NO EXPANSION BOARD PRESENT
INSTALLED	NONE	EXPANSION BOARD PRESENT, 25 WATTS MAXIMUM
NONE	INSTALLED	EXPANSION BOARD PRESENT, 15 WATTS MAXIMUM
INSTALLED	INSTALLED	EXPANSION BOARD PRESENT, 7.5 WATTS MAXIMUM

7

THE THREE LED'S (LED1, LED2 AND LED3) ARE USED TO INDICATE THE STATE OF THE XILINX FPGAS, U10, U11 AND U14 RESPECTIVELY.

* THE LED'S WILL BE 'OFF' IF THE CORRESPONDING FPGA HAS NOT BEEN CONFIGURED.
* THE LED'S WILL BE 'ON' (ILLUMINATED) IF THE CORRESPONDING FPGA HAS BEEN CONFIGURED.



3 SECONDARY SIDE (SOLDER SIDE)

SIZE	CAGE	DWG NO.	REV
B	6X005	PCI-6X00-CI	C
SCALE: NONE		SHEET 2	

REVISIONS				
ZONE	LTR	DESCRIPTION	DATE	APPD

8

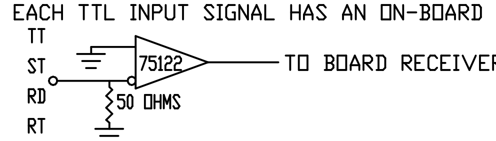
P3 CONNECTOR PIN ASSIGNMENTS

CONNECTOR PIN NO.	SCHEMATIC MNEMONIC	SIGNAL DESCRIPTION
1	REF 422P	REFERENCE SIGNAL; EXTERNAL DDS REFERENCE FROM RS-422 INPUT
2	REF 422N	REFERENCE SIGNAL; EXTERNAL DDS REFERENCE FROM RS-422 INPUT
3	ST 422P	SEND TIMING; EXTERNAL TRANSMIT CLOCK FROM RS-422 INPUT
4	ST 422N	SEND TIMING; EXTERNAL TRANSMIT CLOCK FROM RS-422 INPUT
5	SD SEL*	SEND DATA; TRANSMIT DATA TO THE RS-422 OUTPUT (SD422P OR SD232P)
6	SD 422N*	SEND DATA; TRANSMIT DATA TO THE RS-422 OUTPUT
7	TT SEL	TERMINAL TIMING; TRANSMIT CLOCK RS-422 OUTPUT (TT422P OR TT232P)
8	TT 422N	TERMINAL TIMING; TRANSMIT CLOCK RS-422 OUTPUT
9	RD SEL*	RECEIVE DATA; RECEIVE DATA FROM RS-422 INPUT
10	RD 422N*	RECEIVE DATA; RECEIVE DATA FROM RS-422 INPUT
11	RT SEL	RECEIVE TIMING; RECEIVE CLOCK FROM RS-422 INPUT
12	RT 422N	RECEIVE TIMING; RECEIVE CLOCK FROM RS-422 INPUT
13	REF TTL	REFERENCE SIGNAL; EXTERNAL DDS REFERENCE FROM TTL INPUT
14	ST TTL	SEND TIMING; EXTERNAL TRANSMIT CLOCK FROM TTL INPUT
15	SD TTL*	SEND DATA; TRANSMIT DATA TO TTL OUTPUT
16	TT TTL	TERMINAL TIMING; TRANSMIT CLOCK TTL OUTPUT
17	RD TTL*	RECEIVE DATA; RECEIVE DATA FROM TTL INPUT
18	RT TTL	RECEIVE TIMING; RECEIVE CLOCK FROM TTL INPUT
19	ST EXT	SEND TIMING; CLOCK OUTPUT FROM DIRECT DIGITAL FREQUENCY SYNTHESIZER
20-37	GND	SIGNAL GROUND

NOTE: TTL OUTPUTS ARE OPEN EMITTER AND REQUIRE A 50 OHM LOAD.

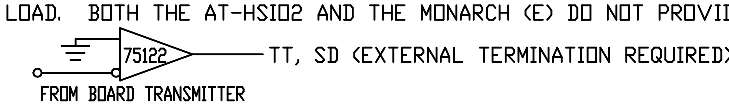
TTL RECEIVER:

THE FOLLOWING SIGNALS ARE APPLICABLE: REF_TT, ST_TTL, RD_TTL, RT_TTL.
THE ASSOCIATED DEVICE ON THE CCA IS: SN75122
HIGH LEVEL ≥ 2.0 VOLTS
LOW LEVEL ≤ 0.8 VOLTS
EACH TTL INPUT SIGNAL HAS AN ON-BOARD 50 OHM TERMINATION.



TTL TRANSMITTER:

THE FOLLOWING SIGNALS ARE APPLICABLE: TT_TTL, SD_TTL.
THE ASSOCIATED DEVICE ON THE CCA IS: SN75121
HIGH LEVEL ≈ 3.0 VOLTS
LOW LEVEL ≈ 0.0 VOLTS
THE SN75121 DEVICES USES EMITTER-FOLLOWER OUTPUTS AND REQUIRES THE LOAD TO BE TERMINATED INTO A 50-500 OHM LOAD. BOTH THE AT-HSIO2 AND THE MONARCH (E) DO NOT PROVIDE ONBOARD TERMINATION FOR THE TT AND SD SIGNALS.



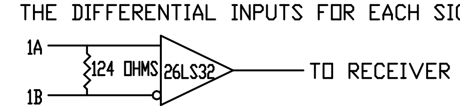
RS-422 RECEIVER:

THE FOLLOWING SIGNALS ARE APPLICABLE: REF_422, ST_422, RD_422, RT_422.
THE ASSOCIATED DEVICE ON THE CCA IS: AM26LS32

INPUT	OUTPUT (TO RECEIVER)
$V_{ID} \geq V_{TH\ (MAX)}$	H
$V_{ID} \leq V_{TH\ (MIN)}$	L

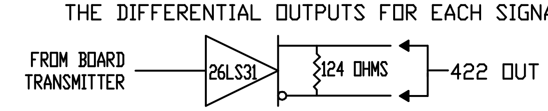
V_{ID} = DIFFERENTIAL INPUT VOLTAGE
V_{TH} = DEVICE THRESHOLD VOLTAGE

COMMON-MODE RANGE = ±7 VOLTS
THE DIFFERENTIAL INPUTS FOR EACH SIGNAL HAS AN ON-BOARD PARALLEL TERMINATION OF 124 OHMS.



RS-422 TRANSMITTER:

THE FOLLOWING SIGNALS ARE APPLICABLE: TT_422, SD_422,
THE ASSOCIATED DEVICE ON THE CCA IS: AM26LS31
HIGH LEVEL ≈ 3.0 VOLTS
LOW LEVEL ≈ 0.0 VOLTS
THE DIFFERENTIAL OUTPUTS FOR EACH SIGNAL HAS AN ON-BOARD PARALLEL TERMINATION OF 124 OHMS.



SIZE	CAGE	DWG NO.	REV
B	6X005	PCI-6X00-CI	C
SCALE: NONE		SHEET 3	

Appendix B:

ACRONYMS AND ABBREVIATIONS

APC_FLG	Autopolarity Correct Flag
ASIC	Application Specific Integrated Circuit
BCTC	Bit Counter Terminal Count
BPWD <3:0>	Bits Per Word
BSWAP	Byte Swap
BURST_EN	Burst Enable
CCSDS	Consultative Committee for Space Data Systems
CLK_ACK	Clock Acknowledge
CLK_FLG	Clock Flag
CONV_EN	Convolutional Encoder Enable
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CRC_EN	CRC Enable
CRC_ERR_FLG	CRC Error Flag
CRC_MODE	CRC Mode
DAC	Digital-to-Analog Converter
DDS	Direct Digital Synthesizer
DDS_EN	Direct Digital Synthesizer Enable
DDSCLK	Direct Digital Synthesizer Clock Register
DMA	Direct Memory Access
EBLEN <7:0>	Error Burst Length
EOF_ACK	Acknowledges End of Frame Interrupt
EOF_FLG	End of Frame Flag
EOF_IEN	End of Frame Interrupt Enable
ERR	Underrun not on FRAME Boundary
ERR_DAT	Error Data Register
FAF_ACK	FIFO Almost Full Acknowledge

FAF_FLG	FIFO Almost Full Flag
FAF_IEN	FIFO Almost Full Interrupt Enable
FIFO	First In First Out
FPGA	Field Programmable Gate Array
I <2:0>	Interleave Level
I/O	Input Output
IDLE_DAT	Idle Data
RSDEF*	Reed-Solomon Decoder FIFO Buffer Empty Flag
RSDFF*	Reed-Solomon Decoder FIFO Buffer Full Flag
RSDHF*	Reed-Solomon Decoder FIFO Buffer Half Full Flag
RSDOVRN_FLAG	Reed-Solomon Decoder Overrun Flag
RSDOVRN_IEN	Reed-Solomon Decoder Overrun Interrupt Enable
INS_ERR	Insert Error
INS_SLIP	Insert Slip
KB	Kilobyte
LOCK_FLG	Lock Flag
LOCK_IEN	Lock Flag Interrupt Enable
LOD	Loss of Data
LOD_ACK	Acknowledges Loss of Data Interrupt
LOD_FLG	Loss of Data Flag
LOD_IEN	Loss of Data Interrupt Enable
LSB	Least Significant Bit/Byte
Mbit/sec	Mega Bit per second
MHz	Megahertz
MSB	Most Significant Bit/Byte
NASA-GSFC	NASA - Goddard Space Flight Center
NASCOM	NASA Communications Network
NRZ-L	Non-Return to Zero – Level
NRZ-M	Non-Return to Zero – Mark
NRZ-S	Non-Return to Zero – Space
OC_TC	Octet Counter Terminal Count
OCD	Octet Counter Data Bits
OCSEL <2:0>	Output Code Select

OCTC_FLG	Octet Counter Terminal Count Flag
PCIOVRN_FLAG	PCI FIFO Overrun Flag
PCIEF*	Serial Input PCI FIFO Buffer Empty Flag
PCIFF*	Serial Input PCI FIFO Buffer Full Flag
PCIHF*	Serial Input PCI FIFO Buffer Half Full Flag
OVRN	Overrun
OVRN_ACK	Overrun Acknowledge
OVRN_FIFO	Overrun FIFO Flag
OVRN_FLG	Overrun Flag
OVRN_IEN	Overrun Interrupt Enable
PATERR	Sync Pattern Error
PC	Personal Computer
PCI	Peripheral Component Interface
PCM	Pulse Code Modulation
PLL	Phase Lock Loop
PREAM <15:0>	This defines the length of the Preamble
PREAM_EN	Preamble Enable
RAND_EN	Randomizer Enable
RAND	Randomizer
RIRQ	Receive Interrupt Request
REF_INV	Inverts Reference Signal
REF_POT	Select Potentiometer Reference
REF_PLL_EN	Reference PLL Enable
REFSEL <1:0>	Reference Select
ROM	Read Only Memory
RS	Reed-Solomon
RS_EN	Reed-Solomon Encoder Enable
RS_OVWR	Reed-Solomon Encoder Parity Overwrite
RSEC	Reed-Solomon Error Correction Chip
RSEC_IRQ	Reed-Solomon Error Correction Chip Interrupt Request
RSERR	Reed-Solomon Encoder Interrupt Request
RSERR_IEN	Reed-Solomon Error Interrupt Enable
SDIS<2:0>	Slip Distance

SLIPDIR	Slip Direction
SLIP_FLG	Slip Flag
SLIP_IEN	Bit Slip Flag Interrupt Enable
SLIPSEL <1:0>	Size of Bit Slip Window
SOEF*	Serial Output FIFO Buffer Empty Flag
SOFF*	Serial Output FIFO Buffer Full Flag
SOHF*	Serial Output FIFO Buffer Half Full Flag
ST_422	Send Timing 422
ST_DDS	Send Timing DSS
ST_INT	Send Timing Internal
ST_PLL	Send Timing PLL
ST_TTL	Send Timing TTL
STCLKA	Transmit Clock Register
ST_PLL_EN	Transmit CLK Enable
STSEL <1:0>	Send Timing Input Selection
SYM_REV	Symbol Reverse
SYMA_INV	Symbol Invert
SYMB_INV	Symbol Invert
TC_INV	Transmit Clock Invert
TD_INV	Transmit Data Invert
TDM	Time Division Multiplex
TIRQ	Transmitter Interrupt Request
TMSV	Time Stamp Overflow
UNLOCK_FLG	Unlock Flag
VCDU	Virtual Channel Data Unit
WCTC	Word Count
WIN_END_FLG	Window End Flag
WIN_END_IEN	Window End Interrupt Enable
XCENC_DONE	Reed-Solomon Encoder FPGA Configuration Complete
XCENC_INIT*	Reed-Solomon Encoder FPGA Initialization
XCENC_PROG	Reed-Solomon Encoder FPGA Program